Textbook Non-Malleable Commitments

Vipul Goyal* Omkant Pandey† Silas Richelson‡

Abstract

We present a new non-malleable commitment protocol. Our protocol has the following features:

- The protocol has only *three rounds* of interaction. Pass (TCC 2013) showed an impossibility result for a two-round non-malleable commitment scheme w.r.t. a black-box reduction to any "standard" intractability reduction. Thus, this resolves the round complexity of non-malleable commitment at least w.r.t. black-box security reductions. Our construction is secure as per the standard notion of non-malleability w.r.t. commitment.
- Our protocol is *truly efficient*. In our basic protocol, the entire computation of the committer is dominated by just three invocations of a non-interactive statically binding commitment scheme, while, the receiver computation (in the commitment stage) is limited to just sampling a random string. Unlike many previous works, we directly construct a protocol for large tags and hence avoid any non-malleability amplification steps.
- Our protocol is based on a black-box use of any non-interactive statistically binding commitment scheme. Such schemes, in turn, can be based on any one-to-one one-way function (or any one-way function at the cost of an extra initialization round). Previously, the best known black-box construction of non-malleable commitments required a larger (constant) number of rounds.
- Our construction is public-coin and makes use of only black-box simulation. Prior to our work, no public-coin constant round non-malleable commitment schemes were known based on black-box simulation.

Our techniques depart *significantly* from the techniques used previously to construct non-malleable commitment schemes. As a main technical tool, we rely on non-malleable codes in the split state model. Our proofs of security are purely combinatorial in nature.

In addition, we also present a simple construction of constant round non-malleable commitments from any one-way function. While this result is not new, the main feature is its simplicity compared to *any* previous construction of non-malleable commitments (in any number of rounds). We believe the construction is simple enough to be covered in a graduate level course on cryptography. The construction uses non-malleable codes in the split state model in a black-box way.

*Microsoft Research, India. Email: vipul@microsoft.com.

†University of California, Berkeley. Email: omkant@berkeley.edu.

‡MIT. Email: SiRichel@csail.mit.edu. Work done in part while visiting Microsoft Research, India.

1 Introduction

Man-in-the-middle (MIM) attacks are one of the most basic attacks in cryptography. The notion of non-malleable commitments was introduced in a seminal work of Dolev, Dwork and Naor [DDN91] as a countermeasure against such adversaries. Since their introduction, non-malleable commitments have proven to be capable of preventing MIM attacks in a variety of settings. Non-malleability lies at the heart of secure protocol composition, it allows for round-efficient secure multi-party computation and gives applications to areas as diverse as position based cryptography [CGMO09].

A *commitment scheme* is a useful two party protocol which allows a committer, $\mathcal{C}$, to send a representation of his message v, $\mathrm{Com}(v;r)$ to a receiver, $\mathcal{R}$, in such a way so that 1) $\mathcal{R}$ learns nothing about v until $\mathcal{C}$ chooses to open his commitment and 2) $\mathcal{C}$ is bound to v; he cannot open $\mathrm{Com}(v)$ to any value $v' \neq v$. A commitment scheme is *non-malleable* if for every message v, no MIM adversary, intercepting a commitment $\mathrm{Com}(v;r)$ and modifying it at will, is able to efficiently generate a commitment $\mathrm{Com}(\tilde{v};\tilde{r})$ to a related message $\tilde{v}$. Interest in non-malleable commitments is motivated both by the central role that they play in securing protocols under composition [CLOS02, LPV09] and by the unfortunate reality that many widely used commitment schemes are actually highly malleable. Indeed, man-in-the-middle (MIM) attacks occur quite naturally when multiple concurrent executions of protocols are allowed, and can be quite devastating.

Since their conception, non-malleable commitment has been studied extensively, and, with increasing success in terms of characterizing its round complexity. The original construction of [DDN91] gave a protocol with logarithmically many rounds. Barak [Bar02] gave a constant round construction based on non-black-box simulation (which was further improved by Pass and Rosen [PR05b]). More recently, constant round protocols for non-malleable commitment with black-box proofs of security were given by Goyal [Goy11] and Lin and Pass [LP11]. Other constructions include [PR05a, LP09, LPV08, PPV08, PW10, Wee10, GLOV12]. The current state of art is represented by the work of Goyal, Richelson, Rosen and Vald [GRRV14] whose scheme requires only four rounds of interaction. On the negative side, Pass [Pas13] showed that two-round non-malleable commitments cannot exists w.r.t. black-box proofs of security based on any "standard" intractability assumption. The lower bound of Pass holds even if the construction uses the underlying assumption in a non-black-box way. Thus, the main remaining open problem regarding the round complexity of non-malleable commitment is

Does there exist a protocol for non-malleable commitment with only three rounds of interaction?

In this work we present a three round protocol, thus giving a positive answer to the above question.

Zero-Knowldge: A Barrier to Three-Round Non-Malleable Commitment. Almost all previous schemes invoke some sort of proof of consistency. These are usually critical to the proofs of non-malleability, as without consistency one runs into a host of "selective $\perp$ attacks" (where the MIM plays in such a way so that whether or not his commitment is valid, depends on the value inside the commitment he receives) which are difficult to rule out. Generally, zero-knowledge is used for this purpose. For example, the recent works of [GRRV14, BGR$^+$15] use a three round "commit-and-prove" sigma protocol along with a GMW-style zero-knowledge proof of correctness [GMW87]. They then use the Feige-Shamir paradigm [FS90] in order to parallelize their protocol down to four rounds.

Zero-knowledge, however, is known to require 4-rounds [GK96] (at least w.r.t. black-box simulation), so if we hope to get three round non-malleable commitment, we must overcome our dependency on zero-knowledge. The main observation which makes this possible is that using zero-knowledge to prove consistency is actually overkill. We do not need to be certain that M has played honestly; we only require that whether M has played honestly or not, cannot depend on $\mathcal{C}$'s commitment v. Capitalizing on this observation, however, is challenging and requires new ideas and a new protocol. Indeed, if we simply remove the zero-knowledge from [GRRV14], the resulting protocol is subject to easy mauling attacks (see the Appendix).

We also point out that also that zero-knowledge is usually the most computationally expensive component in protocols for non-malleable commitment. Indeed, all previous schemes for non-malleable commitment are considerably slower than their ordinary statically binding counterparts.

Our Contributions. We present a new construction of non-malleable commitment which has the following features:

- The protocol has only *three rounds* of interaction. Pass [Pas13] showed that two-round non-malleable commitments unfortunately cannot exists w.r.t. black-box proofs of security based on any "standard" intractability assumption. The lower bound of Pass holds even if the construction uses the underlying assumption in a non-black-box way. Thus, this resolves the round complexity of non-malleable commitments at least w.r.t. black-box security reductions. Our construction is secure as per the standard notion of non-malleability w.r.t. commitment.

- Our protocol is simple and *truly efficient*. In our basic protocol, the entire computation of the committer is dominated by just three invocations of a non-interactive statically binding commitment scheme, while, the receiver computation (in the commitment stage) is limited to just sampling a random string. The decommitment stage is equally basic: the committer would send the openings of these commitments, while, the receiver would be required to check these openings for correctness and perform some simple computations. The protocol is easy to describe, the main complexity lies in the analysis rather than the construction.

  In several previous works (including [GRRV14, BGR$^+$15]), first a non-malleable commitment scheme for "small" tags is constructed. Then, a scheme for large tags is obtained using non-malleability amplification [DDN91, LP09]. This adds a significant multiplicative overhead to the computation of each party: the multiplicative overhead is typically related to the number of bits in the large tags. Unlike these previous works, our basic protocol works directly with large tags, and hence, we avoid any expensive amplification steps. Our basic protocol provides security only against synchronizing adversaries. Extension to non-synchronizing adversaries is addressed later, though still with a three round protocol.

- Our protocol is based on black-box use of any non-interactive statistically binding commitment scheme. Such schemes, in turn, can be based on any one-to-one, one-way function, or, at the cost of an extra initialization round, any one-way function. Previously, the best known black-box construction of non-malleable commitments required a larger constant number of rounds [GLOV12, KMO14]. Furthermore, the previous constructions, even though black-box, were significantly less efficient [GLOV12, LP12, KMO14, Kiy14]. For example, the construction of Goyal et al [GLOV12] used "MPC in the head techniques" of Ishai et. al [IKOS07].

- Our construction is public-coin and makes use of only black-box simulation. Prior to our work, no public-coin constant round non-malleable commitment schemes were known based on black-box simulation. The structure of our basic protocol is arguably "as basic as it can be": $\mathcal{C}$ sends a single commitment to some string, the receiver sends a random challenge, and, in the final round, sender sends another string (but doesn't send any opening).

- Finally, we present a simple — in fact, almost elementary — construction of a constant-round non-malleable commitment scheme using split state non-malleable codes. This construction can be based on any one-way function. While such a construction is not new, the main feature is its simplicity over any previous construction: it has a simple "commit-and-prove" structure and can be be based on any split state non-malleable code (without any additional properties). The reduction makes a standard use of adaptive security, and may be of independent interest. We believe that this protocol is simple enough to be included in a graduate level textbook on cryptography or be taught in a graduate level cryptography course.

Starting Technical Idea. Our key technical tool will be non-malleable codes in the split-state model [DPW10, DKO13]. We will use these codes in conjunction with a techniques introduced by Goyal et. al. [GRRV14]. Non-malleable codes in the split-state model are codes whose codewords are pairs $(\mathsf{L}, \mathsf{R}) \in \mathcal{L} \times \mathcal{R}$, and the tampering function family is

$$\mathcal{F}_{\mathsf{split}} = \big\{(f, g)\big| f : \mathcal{L} \to \mathcal{L}, g : \mathcal{R} \to \mathcal{R}\big\}.$$

That is, one encodes a message by breaking it into two states, and is ensured that non-malleability holds as long as the adversary tampers each state separately. Several recent and exciting works [DKO13, ADL14, CZ14, ADKO15, CGL15] establish a connection between split-state non-malleable codes and various types of randomness extractors. Thus, split-state non-malleable codes allow us to harness deep theorems from the extensive randomness extraction literature and direct them towards the seemingly unrelated area of cryptographic non-malleability.

As a first attempt towards constructing non-malleable commitments, consider the protocol where the committer $\mathcal{C}$ simply commits separately to L and R? This does not work as the underlying commitment scheme may have some homomorphic properties allowing the MIM to maul L and R "jointly". Our starting idea is as follows. Let us focus our attention on synchronizing adversaries[1]. $\mathcal{C}$ encodes the message v as L and R, and, in the first round, sends a commitment $\mathrm{Com}(\mathsf{L})$ to L. The receiver responds back with an acknowledgement message, at which point $\mathcal{C}$ sends R *in the clear*. This scheme does seem to have some non-malleability features. In the first round, the MIM must maul $\mathrm{Com}(\mathsf{L})$ into $\mathrm{Com}(\tilde{\mathsf{L}})$ without knowledge of R, while in the final round, the MIM receives R and must produce $\tilde{\mathsf{R}}$ given only $\mathrm{Com}(\mathsf{L})$ (rather than L itself). While this is indeed our starting point, this intuition turns out to be not sound (see Appendix for an "explicit" attack). Our basic protocol is quite simple and is given below.

- **Committer's Input:** A value v to commit to.
- **1.** $\mathcal{C} \to \mathcal{R}$**:** $\mathcal{C}$ chooses $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(v)$ where L is viewed as a field element in $\mathbb{Z}_q$; $\mathcal{C}$ also draws $r \leftarrow \mathbb{Z}_q$ at random and sends $\mathrm{Com}(\mathsf{L} \circ r)$ to R where Com is a non-interactive, statistically binding commitment scheme.
- **2.** $\mathcal{R} \to \mathcal{C}$**:** $\mathcal{R}$ chooses a random $\alpha \leftarrow \mathbb{Z}_q^*$ and sends it to $\mathcal{C}$.
- **3.** $\mathcal{C} \to \mathcal{R}$**:** $\mathcal{C}$ sends $a = r\alpha + \mathsf{L}$ and R to $\mathcal{R}$.
- **Decommitment:** To decommit, $\mathcal{C}$ decommits to the commitment in **1.**

Intuitively, $\mathcal{C}$ commits to a polynomial-based 2-out-of-2 secret sharing of L in the first round, and in the third round sends R along with one share. The same polynomial based commit-and-reveal mechanism is used in [GRRV14], but no non-malleable codes are used and so a zero-knowledge proof of consistency is needed.

To prove security of the above protocol, we must reduce any "successful" mauling attack to a mauling attack on the underlying non-malleable code. The adversary for the non-malleable codes (*i.e.*, a split-state tampering function pair (f, g)) would have to run the left execution (with the MIM) using the given L and R and extract $\tilde{\mathsf{L}}$ and $\tilde{\mathsf{R}}$ from the right execution. If MIM successfully mauls the commitment scheme, then the extracted $\tilde{\mathsf{L}}$ and $\tilde{\mathsf{R}}$ would decode to $\tilde{v}$ which is related to $\mathcal{C}$'s committed value v represented by L and R. This would presumably contradict the security of the non-malleable code. However one must keep in mind that (f, g) must be split-state and so neither function is allowed to see L and R at once. Hence, it cannot simply run our non-malleable commitment protocol, and extract the tampered $\tilde{\mathsf{L}}$ and $\tilde{\mathsf{R}}$.

To complete the proof of security, we need to construct split-state (f, g) (which can use the MIM and the distinguisher for our protocol internally) so that f outputs $\tilde{\mathsf{L}}$ using only L and likewise g outputs $\tilde{\mathsf{R}}$ using only R. Therefore f and g cannot complete the protocol execution with MIM to extract the tampered code (since each

[1] Roughly, this means that the MIM sends the i-th round message on the right immediately after getting the i-th round message in the left interaction.

will be missing one of L and R)! Thus, the idea of reducing the security of our construction to the security of non-malleable codes (in the split-state model) seems like a non-starter. *This is the key technical challenge we encounter in our proof of non-malleability.*

Our proof strategy, at a very high level is to have $f(\mathsf{L})$ and $g(\mathsf{R})$ execute independent interactions with MIM, and output $\tilde{\mathsf{L}}$ and $\tilde{\mathsf{R}}$, respectively. This, however, leads to $\tilde{\mathsf{L}}$ and $\tilde{\mathsf{R}}$ being extracted from two different protocol transcripts, and so there is no clear way to relate the extracted values back to MIM's mauled commitment $\tilde{v}$. We then show that there exists a single protocol transcript (from the correct distribution) such that the left execution in that transcript is completed using L and R, and the right uses $\tilde{\mathsf{L}}$ and $\tilde{\mathsf{R}}$. Thus, if MIM is successful in mauling the commitment scheme, then (f, g) is split-state and succeeds in mauling the non-malleable code. A more precise technical overview is given later in this section.

We are not able to make the above argument go through based on standard split-state non-malleable codes. We need need the following additional properties described informally below. See Section 3 for a more precise description.

1. The code must be an *augmented* split-state non-malleable code [AAG+16]. This means that the distinguisher for the non-malleable code is given R as input, in addition to the tampered decoded message.

2. We need the code to be *conditionally* non-malleable. Intuitively, this means that non-malleability holds when L is chosen randomly along with (and independently from) the tampering functions (f, g), and R is chosen randomly in the tampering experiment. This is in contrast to the usual security game where (f, g) are fixed and then (L, R) are both drawn during the tampering experiment. We define this new property formally in Section 3.1.

3. The code must satisfy what we call the *simulatable right state* property. Roughly speaking, this means that given a random L, the sets $\{\mathsf{R} : \mathsf{Dec}(\mathsf{L}, \mathsf{R}) = v\}$ and $\{\mathsf{R} : \mathsf{Dec}(\mathsf{L}, \mathsf{R}) = v'\}$ should be indistinguishable for all v, v'. See Section 3.2 for more details.

Our construction is based on the recent split state non-malleable code of Aggarwal et. al [ADL14]. It turns out that the code in [ADL14] already satisfies the first two of the above properties. We then present a modification to add the simulatable right state property. Note that the code of [ADL14] is purely information theoretic, and, in comparison to cryptographic objects (such as commitments or one-way functions), very efficient. Encoding and decoding simply requires sampling random vectors and taking their inner product, etc. To add the strong hiding property, we add a commitment and a symmetric encryption to the encoding and decoding procedures. Thus, our overall basic protocol has computation which is dominated by three invocations of a statistically binding commitment scheme (or rather two invocation of a statistically binding commitment and one symmetric encryption).

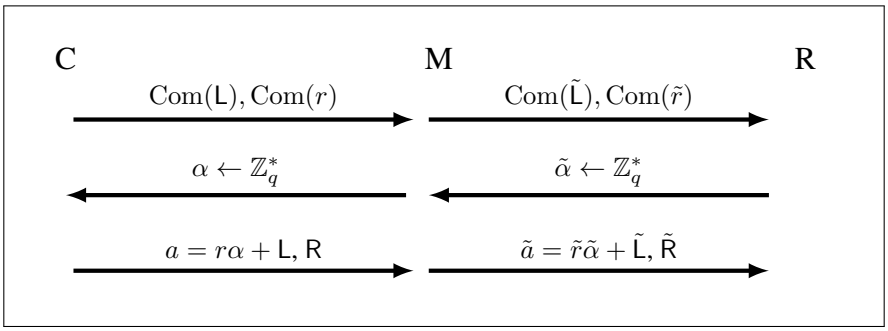

Figure 1: Protocol with Man-in-the-Middle

Overview of the Proof of Non-Malleability. Our protocol (under a man-in-the-middle attack) is given in Figure 1. For the time being, think of $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(v)$ where $\mathsf{L}, \mathsf{R} \in \mathbb{Z}_q$, v is the value to which $\mathcal{C}$ wishes to commit and $(\mathsf{Enc}, \mathsf{Dec})$ is a generic split-state non-malleable code. We will point out where each of the three above additional properties on $(\mathsf{Enc}, \mathsf{Dec})$ is needed. The first is easy to see. The distinguisher for non-malleable commitment gets the transcript (which contains R in the clear) as well as the decommitment $\tilde{v}$, so we need $(\mathsf{Enc}, \mathsf{Dec})$ to be augmented non-malleable.

As mentioned above, we prove non-malleability by reducing any mauling attack to a tampering function (f, g) which mauls the underlying code. This must be done carefully as on the one hand, we need to ensure that f and g tamper v correctly to M's commitment $\tilde{v}$, on the other hand, (f, g) must be split-state, and so cannot naïvely run MIM using both L and R.

Both functions will obtain their output using the third message. They share a partial transcript consisting of the first two messages and the value a. Though this information contains L and so it shouldn't be given to g, L is computationally hidden so we will be able to hybrid it away later. $f(\mathsf{L})$ extracts $\tilde{\mathsf{L}}$ by choosing a random value $\mathsf{R}_\$$ and sending $(a, \mathsf{R}_\$)$, receiving $(\tilde{a}_\$, \cdot)$; then it rewinds M and asks another random challenge $\tilde{\beta}$ on the right, it receives β on the left and sends $(b, \mathsf{R}_\$)$ where $b = (a - \mathsf{L})\big(\beta/\alpha\big) + \mathsf{L}$, and receives $(\tilde{b}, \cdot)$. It outputs $\tilde{\mathsf{L}}$, the constant term on the line spanned by $\big\{(\tilde{\alpha}, \tilde{a}_\$), (\tilde{\beta}, \tilde{b})\big\}$. $g(\mathsf{R})$ also shares the random value $\mathsf{R}_\$$ and so can compute $\tilde{a}_\$$. $g(\mathsf{R})$ rewinds M and sends (a, R) on the left and receives $(\tilde{a}, \tilde{\mathsf{R}})$ on the right. If $\tilde{a} = \tilde{a}_\$$, $g(\mathsf{R})$ outputs $\tilde{\mathsf{R}}$, otherwise it outputs $\perp$.

Note that for (f, g) to succeed in extracting $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$, it must be that the answer $\tilde{a}_\$$ M provides when given the random $\mathsf{R}_\$$ is equal to the $\tilde{a}$ he provides given R. This will follow from the right state simulatability of $(\mathsf{Enc}, \mathsf{Dec})$. Given this property, we can show that the chance that M answers correctly (*i.e.*, consistently with the linear map he committed to in the first round) given $\mathsf{R}_\$$ is about the same as the chance he answers correctly given R. So either both are incorrect with high probability, in which case M is always committing to $\tilde{v} = \perp$ and so cannot be mauling; or else both $\tilde{a}_\$$ and $\tilde{a}$ are correct with non-negligible probability. In this case, we can show that (f, g) succeed in extracting correct $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$ with non-negligible probability. One subtle point is that (f, g) are defined using L. This means that even after we hybrid this dependence away, the mauling experiment for the resulting tampering functions will have L fixed. This is why we need $(\mathsf{Enc}, \mathsf{Dec})$ to be conditionally non-malleable.

Extension to non-synchronizing adversaries. While in some applications, security against synchronizing adversaries is all one needs (e.g., constructing round efficient multi-party computation), in others, non-malleability against arbitrary schedulings is required. Our basic protocol only provides security against synchronizing adversaries, and actually is susceptible to selective bot attacks against general schedulings. Since our protocol has only three rounds, we can enumerate over all possible schedulings and check that the only other potentially problematic scheduling is the sequential one where the left execution finishes entirely even before the right execution starts.

To extend to non-synchronizing adversaries, we make our protocol extractable by running a 3-round extractable (malleable) commitment scheme in parallel to our basic protocol (we do not need any "proofs of consistency" between the two parallel executions). Extraction immediately yields non-malleability against a sequential adversary as we may rewind M and extract his commitment without having to rewind the honest committer. The main technical challenge for this portion is proving non-malleability against a synchronizing adversary; *i.e.*, that the extractable commitment doesn't "interfere" with the basic protocol's synchronizing non-malleability. To achieve this, we will use an extractable commitment scheme such that extraction requires two rewindings instead of just one. This is inspired by a technique from the constant round protocol of Lin and Pass [LP11]. Our final protocol requires significantly more invocations of the underlying commitment scheme, however we stress that even so, it is significantly more efficient than any of the prior ones schemes in the literature (on top of requiring only 3-rounds of interaction). This protocol is described in Section 7.

A simple constant-round non-malleable commitment scheme from any one-way function. We now briefly discuss the ideas behind our simplest protocol. This protocol, and its proof, use only elementary techniques and are simple enough to be included in a graduate level course in cryptography.

The protocol has a simple "commit-and-prove" structure where commitments are executed using Naor's commitment [Nao91]. Recall that Naor's commitment uses a random string ρ sent by the receiver. If ρ has a special form, then the commitment is "equivocal" and can be opened to both 0 and 1; otherwise it is statistically binding. Let us denote this commitment by com_ρ. In out protocol ρ is obtained by simulatable coin-flipping.

Informally, the protocol between the committer $\mathcal{C}$ and receiver $\mathcal{R}$ is as follows:

1. **Sample ρ with coin-tossing:** $\mathcal{C}$ commits to a random string ρ' and sends it to the receiver — this can be done using standard 2-round Naor's commitment. $\mathcal{R}$ responds with a random string ρ''. $\mathcal{C}$ sends $\rho = \rho' \oplus \rho''$ to $\mathcal{R}$ and proves in (constant round) zero-knowledge that ρ is indeed correct. Parse $\rho = \rho_1 \| \rho_2 \| \rho_3$.
2. **Commit the first state:** $\mathcal{C}$ encodes the message m (and identity id) using a standard split-state non-malleable code to obtain two states (L, R). It then commits to L using ρ_1 as the first message of Naor: $c_1 = \mathsf{com}_{\rho_1}(\mathsf{L})$.
3. **Start a proof of consistency.** Before sending the second state, $\mathcal{C}$ and $\mathcal{R}$ start a zero-knowledge proof-of-knowledge of "consistency" in which the statement to be proven is decided in the last step.

   This is done by using the (standard) Feige-Shamir ZK protocol [FS89] which has constant rounds. Parties exchange *all but the last* message of this ZK protocol. Each commitment of this ZK protocol is implemented using a unique part of ρ_2 (as first message of Naor).
4. **Send second state and finish the proof.** $\mathcal{C}$ now just need to send the second state R and prove that it is consistent with the value in c_1. We do this slightly differently: $\mathcal{C}$ commits to R using the third part ρ_3, i.e., $c_2 = \mathsf{com}_{\rho_3}(\mathsf{R})$ and proves that (c_1, c_2) are commitments to consistent states by sending the last message of the proof. It then *opens* c_2 to R by sending appropriate decommitment. $\mathcal{R}$ simply checks that the opening of c_2 to R is valid and the proof verifies before accepting the commitment.

The proof crucially relies on the fact that Naor's commitment is equivocal if ρ can be set appropriately. Specifically, we first "cheat" in the first step where ρ is set to be the special string which allows equivocation. This is done by using the simulator of zero-knowledge proof in the first step. For now, assume that the adversary is synchronous. This step ensures that there is no information about the states in the commitments or the proof transcripts.

We now construct split state functions by relying on the equivocality of Naor's commitment. Specifically, our functions will share the same random tape ϕ and will be denoted by (f_ϕ, g_ϕ) for every ϕ. They will both sample a transcript using random values for the states (instead of actual L and R) up to the point where last message is to be sent. At this point, they will output corrupted states differently, as follows.

The function f_ϕ, upon receiving L, will use the equivocality of com w.r.t. strings ρ_1, ρ_2, ρ_3 to find appropriate randomness that is consistent with L and the transcript and construct a "honest committer" algorithm. It will then extract $\widetilde{\mathsf{L}}$ from this machine by rewinding in the proof-of-knowledge part. During rewidings, it suffices to use random (incorrect) values for R; it can be shown that as long as correct L is used in the extraction, we will obtain an appropriately distributed value for $\widetilde{\mathsf{L}}$ on right.

The function g_ϕ, upon receiving R, will compute the last message of the proof just like function f_ϕ. However, it will open c_2 to the correct value R and feed it to MIM. If it receives a valid last message on the right with a state $\widetilde{\mathsf{R}}$, it will output that state (and $\perp$) otherwise.

We remark that even though the transcripts and the commitments generated in the description of (f_ϕ, g_ϕ) are not binding, the proof ensures that the values output by them are distributed appropriately as long as correct states are given to them as input. The asynchronous adversary is handled using a simple scheduling argument (without relying on non-malleable code). We present the full protocol and proof in Section 8.

2 Preliminaries

We use λ for the security parameter and $\mathsf{negl}(\lambda)$ or negl for a function which tends to zero faster than λ^{-k} for any constant k. We say that two probability distributions X and Y are computationally indistinguishable, and write $\mathsf{X} \approx_{\mathsf{c}} \mathsf{Y}$ if for all probabilistic polynomial time (PPT) distinguishers D,

$$\left|\Pr_{x \leftarrow \mathsf{X}}\big(\mathrm{D}(x) = 1\big) - \Pr_{y \leftarrow \mathsf{Y}}\big(\mathrm{D}(y) = 1\big)\right| = \mathsf{negl}.$$

2.1 Cryptographic Building Blocks

Symmetric Key Encryption. A symmetric key encryption scheme is a tuple of algorithms $(\mathsf{G}, \mathsf{E}, \mathsf{D})$: the key generation algorithm G takes input 1^λ and outputs a private key k; the encryption algorithm E takes a key k, a plaintext message msg and randomness r as input and produces a ciphertext $\mathsf{ct} = \mathsf{E}_k(\mathsf{msg}; r)$; and decryption takes a key k and a ciphertext ct and outputs $\mathsf{msg} = \mathsf{D}_k(\mathsf{ct})$. We require

- **Correctness:** For any message msg, the process: $k \leftarrow \mathsf{G}(1^\lambda)$, $\mathsf{ct} \leftarrow \mathsf{E}_k(\mathsf{msg})$, $\mathsf{msg}' = \mathsf{D}_k(\mathsf{ct})$ outputs $\mathsf{msg}' = \mathsf{msg}$ with probability 1.
- **Semantic Security:** For any messages $\mathsf{msg}, \mathsf{msg}'$, we have

$$\big\{\mathsf{ct} : k \leftarrow \mathsf{G}(1^\lambda), \mathsf{ct} \leftarrow \mathsf{E}_k(\mathsf{msg})\big\} \approx_{\mathsf{c}} \big\{\mathsf{ct}' : k \leftarrow \mathsf{G}(1^\lambda), \mathsf{ct}' \leftarrow \mathsf{E}_k(\mathsf{msg}')\big\}.$$

It is known how to construct symmetric key encryption schemes from any one-way function.

Commitment Schemes. A commitment scheme, $\langle \mathcal{C}, \mathcal{R} \rangle$ is a two-phase, two party protocol between a committer $\mathcal{C}$ and a receiver $\mathcal{R}$. In the commit phase, $\mathcal{C}$ uses secret input v and interacts with $\mathcal{R}$ who uses no input. Let $z = \mathrm{Com}(v; r)$ denote $\mathcal{R}$'s view after the commit phase. Let $(w, v) = \mathrm{Decom}(z, v, r)$ denote $\mathcal{R}$'s view after the decommit phase, which $\mathcal{R}$ either accepts or rejects. We say that $\langle \mathcal{C}, \mathcal{R} \rangle$ is a statistically binding commitment scheme if the following properties hold:

- **Correctness:** If parties follow the protocol, then $\mathsf{R}(z, w, v) = 1$;
- **Binding:** With high probability over R's randomness, there does not exist a (w', v') with $v' \neq v$ such that $\mathsf{R}(z, w', v') = 1$;
- **Hiding:** For all v, v', $\big\{\mathrm{Com}(v; r)\big\}_r \approx_c \big\{\mathrm{Com}(v'; r')\big\}_{r'}$.

It is known how to construct a non-interactive, perfectly binding commitment scheme from any one-way permutation [Blu81]. Alternatively, any one-way function can be used to construct a two-round, statistically binding commitment scheme, where the binding property holds with high probability over $\mathcal{R}$'s randomness [Nao91].

Id-based Commitment Scheme. Following [PR05b, DDN91], we consider id-based commitment schemes where, in addition to the security parameter, the committer and the receiver also receive an identity $\mathsf{id} \in \{0,1\}^\lambda$ as common input.

2.2 Non-malleable commitments

Non-malleable commitment is defined using the real/ideal paradigm. In the real interaction, there is a man-in-the-middle adversary M interacting with a committer, $\mathcal{C}$, in the left session a receiver $\mathcal{R}$ in the right. We denote the various quantities associated with the right interaction as "tilde'd" versions of their left counterparts. So for example, $\mathcal{C}$ commits to v in the left interaction while M commits to $\tilde{v}$ in the right. Let $\mathbf{MIM}_v$ denote a random

variable that describes $(\mathsf{VIEW}, \tilde{v})$, consisting of M's view in the experiment and the value M commits to in the right interaction, given that $\mathcal{C}$ has committed to v in the left interaction. The ideal interaction is the same, except that $\mathcal{C}$ commits to an arbitrary fixed value, say 0, on the left. Let $\mathbf{MIM}_0$ be the random variable describing $(\mathsf{VIEW}, \tilde{v})$ in this interaction. We will use id-based commitment schemes and we force M to use an identity $\tilde{\mathsf{id}}$ on the right which is distinct from id used on the left. We enforce this by stipulating that $\mathbf{MIM}_v$ and $\mathbf{MIM}_0$ output the special symbol $\perp_{\mathsf{id}}$ when M has used the same identity on the right which he has received on the left. This is analogous to the uninteresting case when M is simply acting as a channel, forwarding messages from $\mathcal{C}$ to $\mathcal{R}$ and back. We let $\mathbf{MIM}_v(y)$ and $\mathbf{MIM}_0(y)$ be the distributions where M gets a string y as auxiliary input.

Definition 1 (Non-Malleable Commitments). *A commitment scheme $\langle \mathcal{C}, \mathcal{R} \rangle$ is* non-malleable *if for every PPT man-in-the-middle adversary M, and for all v, we have $\{\mathbf{MIM}_v(y)\}_{y\in\{0,1\}^*} \approx_{\mathsf{c}} \{\mathbf{MIM}_0(y)\}_{y\in\{0,1\}^*}$.*

On the Presence of Identities. In the real world, nothing can be done to prevent a MIM from simply acting as a channel between $\mathcal{C}$ and $\mathcal{R}$, forwarding messages back and forth. Therefore, the most basic definition of non-malleable commitment requires that the MIM cannot distinguish whether he is in the real or ideal world, even when he is given his committed value on the right, as long as he is doing something (anything) other than copying. [DDN91] noticed that this is equivalent to requiring every committer to have a public identity and forcing the MIM to use an identity which is distinct from $\mathcal{C}$'s. This observation has persisted ever since, so non-malleable commitment is usually defined with respect to identities. The equivalence is based on the following observation: $\mathcal{C}$ can choose his identity to be the public verification key for a signature and send a signature of the entire transcript in the last round. The security of the signature scheme then ensures that either M has copied $\mathcal{C}$'s identity, in which case he cannot produce a valid signature unless he has copied every message as well, or he has used a distinct identity.

2.3 Non-Malleable Codes

A *coding scheme* is a pair of functions $(\mathsf{Enc}, \mathsf{Dec})$ where $\mathsf{Enc} : \mathcal{M} \to \mathcal{C}$ and $\mathsf{Dec} : \mathcal{C} \to \mathcal{M}$ for a message space $\mathcal{M}$ and codeword space $\mathcal{C}$. It should be the case that $\mathsf{Dec} \circ \mathsf{Enc}(m) = m$ for all $m \in \mathcal{M}$ with high probability over the randomness of Enc (it needn't be the case that Enc is randomized at all, in which case correctness requires $\mathsf{Dec} \circ \mathsf{Enc}(m) = m$ with probability 1). Historically, coding schemes are usually designed in order to be resilient to some form of tampering. In their important 2010 paper Dziembowski, Pietrzak and Wichs [DPW10] introduced non-malleable codes which are codes with strong security in the presence of tampering. Informally, for a family $\mathcal{F} \subset \{f : \mathcal{C} \to \mathcal{C}\}$, we say that $(\mathsf{Enc}, \mathsf{Dec})$ is non-malleable with respect to $\mathcal{F}$ if for all $f \in \mathcal{F}$, the tamper distribution $\big(\mathsf{Dec} \circ f \circ \mathsf{Enc}\big)(m)$ (over the randomness of Enc) outputs $\tilde{m}$ which is either equal to m if f copying or else is independent of m. In this work we are interested in split-state non-malleable codes.

Let $(\mathsf{Enc}, \mathsf{Dec})$ be a split state coding scheme so $\mathsf{Enc} : \mathcal{M} \to \mathcal{L} \times \mathcal{R}$ and let

$$\mathcal{F}_{\mathsf{split}} = \big\{(f, g)\big| f : \mathcal{L} \to \mathcal{L}, g : \mathcal{R} \to \mathcal{R}\big\}$$

be the set of split-state tampering functions.

Definition 2 (Tampering Distribution). *Fix $m \in \mathcal{M}$, $(f, g) \in \mathcal{F}_{\mathsf{split}}$. The* tampering distribution*, denoted $\mathsf{T}_{m,f,g}$ is: draw $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m)$, set $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$ and output $\tilde{m} = \mathsf{Dec}(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$.*

Definition 3 (Simulatable Distribution). *Let $\{\mathsf{D}_m\}_{m\in\mathcal{M}}$ be a family of distributions on $\mathcal{M}$ indexed by m. We say that $\{\mathsf{D}_m\}$ is $\varepsilon-$*simulatable *if there exists a distribution S on $\mathcal{M} \cup \{\mathsf{same}\}$ such that $\Delta(\mathsf{D}_m, \mathsf{S}_m) < \varepsilon$ for all m, where S_m is the distribution on $\mathcal{M}$ induced by drawing $\tilde{m} \leftarrow \mathsf{S}$ and outputting m if $\tilde{m} = \mathsf{same}$, $\tilde{m}$ if not.*

Definition 4 (Split-State Non-Malleable Code). *We say that $(\mathsf{Enc}, \mathsf{Dec})$ is $\varepsilon-$*non-malleable against $\mathcal{F}_{\mathsf{split}}$ *if for all $(f, g) \in \mathcal{F}_{\mathsf{split}}$, $\{\mathsf{T}_{m,f,g}\}_m$ is $\varepsilon-$simulatable.*

The Non-Malleable Code of [ADL14]. The code of [ADL14] maps $m \in \mathcal{M}$ into (L, R) where $\mathsf{L}, \mathsf{R} \in \mathbb{Z}_p^n$ are random subject to the condition that $\langle \mathsf{L}, \mathsf{R} \rangle \in \mathsf{H}_m \subset \mathbb{Z}_p$ (p is a prime much larger than $|\mathcal{M}|$, and the $\{\mathsf{H}_m\}_{m \in \mathcal{M}}$ are carefully chosen disjoint subsets of $\mathbb{Z}_p$). Non-malleability follows from an extensive analysis of the inner product function which makes heavy use of its properties as a randomness extractor. For any $(f, g) \in \mathcal{F}_{\mathsf{split}}$ and $x \in \mathbb{Z}_p$, the following random process is considered: choose $\mathsf{L}, \mathsf{R} \in \mathbb{Z}_p^n$ randomly such that $\langle \mathsf{L}, \mathsf{R} \rangle = x$, set $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$, and output $\tilde{x} = \langle \tilde{\mathsf{L}}, \tilde{\mathsf{R}} \rangle$. The main lemma of [ADL14] says that $\tilde{x}$ is either 1) independent of x, or 2) of the form $\tilde{x} = ax + b$ for some $a, b \in \mathbb{Z}_p$ which depend only on (f, g). Non-malleability in [ADL14] then follows from the design of *affine evasive sets* as the $\{\mathsf{H}_m\}_m$. This aspect of their construction is very elegant but as we will not need to change their $\{\mathsf{H}_m\}_m$, we do not discuss this portion further. The interested reader should see [ADL14] for more information. We note that the earlier work of [DKO13] used essentially the same outline in order to give a non-malleable code for one bit messages. Their construction is also very elegant and is much simpler: they use $\mathsf{H}_0 = \{0\}$ and $\mathsf{H}_1 = \mathbb{Z}_p - \{0\}$.

2.4 Augmented Non-Malleable Codes

Very recently Aggarwal et al. [AAG+16] proved that the [ADL14] construction is non-malleable even when the tamper distribution outputs $\tilde{m}$ along with one of the states. Their proof looks at the randomized process: choose $\mathsf{L}, \mathsf{R} \in \mathbb{Z}_p^n$ randomly such that $\langle \mathsf{L}, \mathsf{R} \rangle = x$, set $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$ and output $(\mathsf{R}, \tilde{x})$ where $\tilde{x} = \langle \tilde{\mathsf{L}}, \tilde{\mathsf{R}} \rangle$. The same randomness extraction properties of the inner product function used in [ADL14] show that even conditioned on R, $\tilde{x}$ is either independent of x or else $\tilde{x} = ax + b$ for $a, b \in \mathbb{Z}_p$ which depend only on (f, g). They call this stronger notion augmented non-malleability.

Definition 5 (Augmented Tampering Distribution). *Fix $m \in \mathcal{M}$ and $(f, g) \in \mathcal{F}_{\mathsf{split}}$. The* augmented tampering distribution*, denoted $\mathsf{V}_{m,f,g}$ is: draw $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m)$, set $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$ and output $(\mathsf{R}, \tilde{m})$ where $\tilde{m} = \mathsf{Dec}(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$.*

We use the letter V for "view": the output of the augmented tampering distribution will be basically what the MIM sees during a mauling attack on our non-malleable commitment scheme.

Definition 6 (Augmented Simulatable Distribution). *Let $\{\mathsf{D}_m\}_{m \in \mathcal{M}}$ be a family of distributions on $\mathcal{R} \times \mathcal{M}$ indexed by m, where $\mathcal{R}$ is an arbitrary set. We say that $\{\mathsf{D}_m\}$ is* $\varepsilon-$augmented simulatable *if there exists a distribution S on $\mathcal{R} \times (\mathcal{M} \cup \{\mathsf{same}\})$ such that $\Delta(\mathsf{D}_m, \mathsf{S}_m) < \varepsilon$ for all m, where S_m is the distribution on $\mathcal{M}$ induced by drawing $(\mathsf{R}, \tilde{m}) \leftarrow \mathsf{S}$ and outputting (R, m) if $\tilde{m} = \mathsf{same}$, $(\mathsf{R}, \tilde{m})$ if not.*

Definition 7 (Augmented Non-Malleable Code). *We say that $(\mathsf{Enc}, \mathsf{Dec})$ is* $\varepsilon-$augmented non-malleable *against $\mathcal{F}_{\mathsf{split}}$ if for all $(f, g) \in \mathcal{F}_{\mathsf{split}}$, $\{\mathsf{V}_{m,f,g}\}_m$ is $\varepsilon-$augmented simulatable.*

Before moving on, we remark that the proof in [AAG+16] actually shows something slightly stronger. Let us think of the randomized process above as drawing $\mathsf{L} \leftarrow \mathbb{Z}_p^n$ at random and then drawing $\mathsf{R} \leftarrow \{\mathbf{v} \in \mathbb{Z}_p^n : \langle \mathsf{L}, \mathbf{v} \rangle = x\}$, computing $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$ and outputting $(\mathsf{R}, \tilde{x})$. The analysis of [AAG+16] does not actually require L to be uniform in $\mathbb{Z}_p^n$ and works whenever L has sufficient min-entropy. In particular, given $n, p, \mathcal{M}, \{\mathsf{H}_m\}_m$ as in [ADL14] and any sufficiently large $\mathcal{L} \subset \mathbb{Z}_p^n$, define the coding scheme:

- $\underline{\mathsf{Enc}^{\mathcal{L}}(m)\text{:}}$ choose $\mathsf{L} \leftarrow \mathcal{L}$ and $\mathsf{R} \leftarrow \mathbb{Z}_p^n$ randomly such that $\langle \mathsf{L}, \mathsf{R} \rangle \in \mathsf{H}_m$.
- $\underline{\mathsf{Dec}(\mathsf{L}, \mathsf{R})\text{:}}$ if $\langle \mathsf{L}, \mathsf{R} \rangle \in \mathsf{H}_m$, output m, otherwise output $\perp$.

Claim 1. *For all $\mathcal{M}$, there exist $n, p = \mathsf{poly}(|\mathcal{M}|, \lambda)$ such that for all $\mathcal{L} \subset \mathbb{Z}_p^n$ of size at least $|\mathcal{L}| \geq p^{\eta n}$, $(\mathsf{Enc}^{\mathcal{L}}, \mathsf{Dec})$ is $2^{-\Omega(\lambda)}-$augmented non-malleable, where $1 - \eta = c \log^{-6} p$ for an absolute constant $c > 0$.*

Moreover, the simulator for $(\mathsf{Enc}^{\mathcal{L}}, \mathsf{Dec})$ is identical to the simulator for $(\mathsf{Enc}, \mathsf{Dec})$ except that it draws $\mathsf{L} \leftarrow \mathcal{L}$ instead of $\mathsf{L} \leftarrow \mathbb{Z}_p^n$. Claim 1 follows from the proof of the main theorem in [AAG+16]; as pointed out to us by Aggarwal in a personal communication [Agg].

3 New Constructions of Non-Malleable Codes

3.1 Conditional Augmented Non-Malleable Codes

Let $(\mathsf{Enc}, \mathsf{Dec})$ be a split-state code with codeword space $\mathcal{L} \times \mathcal{R}$. In proving that our commitment scheme is non-malleable, we will need to choose a random $\mathsf{L} \in \mathcal{L}$ and be ensured that the augmented tampering distribution is independent of m even conditioned on L. We define information theoretic and computational variants of these codes.

Definition 8 (Conditional Augmented Tampering Distribution). *Fix $m \in \mathcal{M}$, $(f, g) \in \mathcal{F}_{\mathsf{split}}$, and $\mathsf{L} \in \mathcal{L}$. The* conditional augmented tampering distribution*, $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ is: draw $\mathsf{R} \leftarrow \mathsf{Enc}(m|\mathsf{L}) = \{\mathsf{R}' \in \mathcal{R} : \mathsf{Dec}(\mathsf{L}, \mathsf{R}') = m\}$, set $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$ and output $(\mathsf{R}, \tilde{m})$ where $\tilde{m} = \mathsf{Dec}(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$.*

Definition 9 (Conditional Augmented Simulatable Distribution). *Let $\big\{\mathsf{D}^{\mathsf{L}}_m\big\}_{m,\mathsf{L}}$ be a family of distributions on $\mathcal{R} \times \mathcal{M}$ indexed by $m \in \mathcal{M}$ and $\mathsf{L} \in \mathcal{L}$, where $\mathcal{L}$ and $\mathcal{R}$ are arbitrary sets. We say that $\big\{\mathsf{D}^{\mathsf{L}}_m\big\}_{m,\mathsf{L}}$ is* $\varepsilon-$conditionally augmented simulatable *if there exists a family of distributions $\big\{\mathsf{S}^{\mathsf{L}}\big\}_{\mathsf{L}}$ on $\mathcal{R} \times \big(\mathcal{M} \cup \{\mathsf{same}\}\big)$ such that for all (computationally unbounded) distinguishers D,*

$$\Pr_{\mathsf{L}}\left[\exists\, m \in \mathcal{M} \text{ st } \left|\Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{D}^{\mathsf{L}}_m}\big(\mathrm{D}(\mathsf{R}, \tilde{m}) = 1\big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{S}^{\mathsf{L}}_m}\big(\mathrm{D}(\mathsf{R}, \tilde{m}) = 1\big)\right| > \varepsilon\right] < \varepsilon.$$

where the probability is over $\mathsf{L} \leftarrow \mathcal{L}$ drawn uniformly and where $\mathsf{S}^{\mathsf{L}}_m$ draws $(\mathsf{R}, \tilde{m}) \leftarrow \mathsf{S}^{\mathsf{L}}$ and outputs (R, m) if $\tilde{m} = \mathsf{same}$, $(\mathsf{R}, \tilde{m})$ if not. We say $\{\mathsf{D}^{\mathsf{L}}_m\}_{m,\mathsf{L}}$ is computationally conditionally augmented simulatable *if for all PPT distinguishers D and non-negligible $\delta > 0$, there exist simulators $\big\{\mathsf{S}^{\mathsf{L}}\big\}_{\mathsf{L}}$ such that*

$$\Pr_{\mathsf{L}}\left[\exists\, m \in \mathcal{M} \text{ st } \left|\Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{D}^{\mathsf{L}}_m}\big(\mathrm{D}(\mathsf{R}, \tilde{m}) = 1\big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{S}^{\mathsf{L}}_m}\big(\mathrm{D}(\mathsf{R}, \tilde{m}) = 1\big)\right| > \delta\right] = \mathsf{negl}.$$

Definition 10 (Conditional Augmented Non-Malleable Code). *We say that $(\mathsf{Enc}, \mathsf{Dec})$ is $\varepsilon-$conditionally augmented non-malleable (resp. computationally conditionally augmented non-malleable) against $\mathcal{F}_{\mathsf{split}}$ if for all $(f, g) \in \mathcal{F}_{\mathsf{split}}$, $\big\{\mathsf{V}^{\mathsf{L}}_{m,f,g}\big\}_{m,\mathsf{L}}$ is $\varepsilon-$conditionally augmented simulatable (resp. computationally conditionally augmented simulatable).*

We use a simple probability argument to show that the code from [ADL14] is conditionally augmented non-malleable. This is reminiscent of the way in which one argues that a sufficiently good two-source extractor is also a strong two-source extractor.

Claim 2. *The code $(\mathsf{Enc}, \mathsf{Dec})$ of [ADL14] is $\varepsilon'-$conditionally augmented non-malleable for some negligible quantity $\varepsilon' > 0$.*

Proof. Given $(f, g) \in \mathcal{F}_{\mathsf{split}}$, the simulator $\mathsf{S}_{f,g}$ guaranteed by the augmented non-malleability of $(\mathsf{Enc}, \mathsf{Dec})$ behaves as follows: draw $\mathsf{L}, \mathsf{R} \leftarrow \mathbb{Z}_p^n$, set $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$ and output $\big(\mathsf{R}, \langle \tilde{\mathsf{L}}, \tilde{\mathsf{R}} \rangle\big)$ unless $\langle \tilde{\mathsf{L}}, \tilde{\mathsf{R}} \rangle = \langle \mathsf{L}, \mathsf{R} \rangle$, in which case output $(\mathsf{R}, \mathsf{same})$. We define the family $\{\mathsf{S}^{\mathsf{L}}_{f,g}\}_{\mathsf{L}}$ of simulators similarly: $\mathsf{S}^{\mathsf{L}}_{f,g}$ draws $\mathsf{R} \leftarrow \mathbb{Z}_p^n$ at random and outputs $\big(\mathsf{R}, \langle \tilde{\mathsf{L}}, \tilde{\mathsf{R}} \rangle\big)$ or $(\mathsf{R}, \mathsf{same})$ according to whether $\langle \tilde{\mathsf{L}}, \tilde{\mathsf{R}} \rangle$ is distinct from or equal to $\langle \mathsf{L}, \mathsf{R} \rangle$. For a distinguisher D, let

$$\mathcal{L}^{\mathrm{D}}_{\mathsf{bad}} = \Big\{\mathsf{L} \in \mathbb{Z}_p^n : \exists\, m \in \mathcal{M} \text{ st } \Big|\Pr\big(\mathrm{D}(\mathsf{V}^{\mathsf{L}}_{m,f,g}) = 1\big) - \Pr\big(\mathrm{D}(\mathsf{S}^{\mathsf{L}}_{m,f,g}) = 1\big)\Big| > \varepsilon'\Big\},$$

where $\varepsilon' > \zeta = p^{-(1-\eta)n}$ for $1 - \eta = c \log^{-6} p$ for an absolute constant $c > 0$. If $|\mathcal{L}^{\mathrm{D}}_{\mathsf{bad}}| < \zeta p^n$ then we are done so assume $|\mathcal{L}^{\mathrm{D}}_{\mathsf{bad}}| \geq \zeta p^n$. By Claim 1, the restricted code $(\mathsf{Enc}^{\mathcal{L}^{\mathrm{D}}_{\mathsf{bad}}}, \mathsf{Dec})$ is $2^{-\Omega(\lambda)}-$augmented non-malleable with simulator $\mathsf{S}^{\mathcal{L}^{\mathrm{D}}_{\mathsf{bad}}}_{f,g}$ identical to that for $(\mathsf{Enc}, \mathsf{Dec})$ except that the initial choices of L, R are $\mathsf{L} \leftarrow \mathcal{L}_{\mathsf{bad}}$, $\mathsf{R} \leftarrow \mathbb{Z}_p^n$. However, this is a contradiction since by definition of $\mathcal{L}^{\mathrm{D}}_{\mathsf{bad}}$, $\mathsf{S}^{\mathsf{L}}_{f,g}$ does not simulate $\big\{\mathsf{V}^{\mathsf{L}}_{m,f,g}\big\}_m$. □

3.2 Adding the Hiding Property

We will also need our non-malleable code to have a computational hiding property resembling semantic security in order to rule out selective $\perp$ attacks. We formalize the property we need using a game between a challenger $\mathcal{C}$ and a PPT adversary $\mathcal{A}$, parametrized by $N = \mathsf{poly}(\lambda)$, a message $m \in \mathcal{M}$ and a distribution $\mathcal{D}_{\mathsf{hid}}$ on $\mathcal{R}$.

- $\mathcal{C}$ draws $\mathsf{L} \leftarrow \mathcal{L}$, $b \leftarrow \{0,1\}$, sends $\mathsf{R}_1, \ldots, \mathsf{R}_N$ to $\mathcal{A}$ where $\mathsf{R}_i \leftarrow \mathsf{Enc}(m|\mathsf{L})$ if $b = 0$; $\mathsf{R}_i \leftarrow \mathcal{D}_{\mathsf{hid}}$ if $b = 1$.
- $\mathcal{A}$ outputs b' and wins if $b' = b$.

Definition 11 (Codes with Simulatable State). *We say that a split-state code* $(\mathsf{Enc}, \mathsf{Dec})$ *has* simulatable right state *if there is a distribution* $\mathcal{D}_{\mathsf{hid}}$ *on* $\mathcal{R}$ *such that for all PPT* $\mathcal{A}$, $N = \mathsf{poly}(\lambda)$, *and* $m \in \mathcal{M}$, *the probability that* $\mathcal{A}$ *wins the above game is at most* $1/2 + \mathsf{negl}$.

Hiding Game Variant. We will use another game parametrized by polynomials $N, N' = \mathsf{poly}(\lambda)$. In this game $\mathcal{A}$ sends $\mathcal{C}$ two messages $m, m' \in \mathcal{M}$, $\mathcal{C}$ draws a secret $\mathsf{L} \leftarrow \mathcal{L}$ and sends $\mathcal{A}$ the tuple $\big(\mathsf{R}, \{\mathsf{R}_1\}, \ldots, \{\mathsf{R}_N\}\big)$ where $\mathsf{R} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ and each set has N' elements. Moreover, $\mathsf{R}_i \leftarrow \mathsf{Enc}(m'|\mathsf{L})$ for all $\mathsf{R}_i \in \{\mathsf{R}_i\}$ and all $i = 1, \ldots, N$ except for one random i^* for which $\mathsf{R}_{i^*} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ for all $\mathsf{R}_{i^*} \in \{\mathsf{R}_{i^*}\}$; $\mathcal{A}$ tries to guess i^*. If $(\mathsf{Enc}, \mathsf{Dec})$ has simulatable right state then a PPT adversary $\mathcal{A}$ can guess i^* with probability at most $1/N + \mathsf{negl}$.

Construction. Let $(\mathsf{Enc}_0, \mathsf{Dec}_0)$ be an $\varepsilon-$conditional augmented non-malleable code. Let $(\mathsf{G}, \mathsf{E}, \mathsf{D})$ be a symmetric key encryption scheme, and let $(\mathrm{Com}, \mathrm{Decom})$ be a non-interactive, perfectly binding commitment scheme. The new coding scheme, $(\mathsf{Enc}, \mathsf{Dec})$ is defined as follows.

- $\mathsf{Enc}(m)$**:** Draw $(\mathsf{L}_0, \mathsf{R}_0) \leftarrow \mathsf{Enc}_0(m)$, $k \leftarrow \mathsf{G}(1^\lambda)$, $\sigma \leftarrow \$$, and $c \leftarrow \mathsf{E}_k(\mathsf{R}_0)$. Set $z = \mathrm{Com}(k, \sigma)$ and output (L, R) where $\mathsf{L} = \big(\mathsf{L}_0, (k, \sigma)\big)$, $\mathsf{R} = (c, z)$.
- $\mathsf{Dec}(\mathsf{L}, \mathsf{R})$**:** If either $\mathsf{L}, \mathsf{R} = \perp_{\mathsf{com}}$ output $\perp_{\mathsf{com}}$. Otherwise, parse $\mathsf{L} = \big(\mathsf{L}_0, (k, \sigma)\big)$ and $\mathsf{R} = (c, z)$, check that $\mathrm{Decom}(z) = (k, \sigma)$. If so set $\mathsf{R}_0 = \mathsf{D}_k(c)$, output $\mathsf{Dec}_0(\mathsf{L}_0, \mathsf{R}_0)$; if not output $\perp_{\mathsf{com}}$.

Claim 3. $(\mathsf{Enc}, \mathsf{Dec})$ *has simulatable right state.*

Proof. Define three challengers $\mathcal{C}_0$, $\mathcal{C}_1$, $\mathcal{C}_2$ for the above hiding game, played with some fixed $m \in \mathcal{M}$. Each challenger draws $\mathsf{L} = \big(\mathsf{L}_0, (k, \sigma)\big) \leftarrow \mathcal{L}$ and for $i = 1, \ldots, N$

- $\mathcal{C}_0$ sets $\mathsf{R}_i = (c_i, z)$ where $z = \mathrm{Com}(k, \sigma)$ and $c_i \leftarrow \mathsf{E}_k(\mathsf{R}_0)$ for some R_0 with $\mathsf{Dec}_0(\mathsf{L}_0, \mathsf{R}_0) = m$.
- $\mathcal{C}_1$ sets $\mathsf{R}_i = (c_i, z)$ where $z = \mathrm{Com}(0)$ and $c_i \leftarrow \mathsf{E}_k(\mathsf{R}_0)$.
- $\mathcal{C}_2$ sets $\mathsf{R}_i = (c_i, z)$ where $z = \mathrm{Com}(0)$ and $c_i \leftarrow \mathsf{E}_{k'}(0)$, where $k' \leftarrow \mathsf{G}(1^\lambda)$.

Note that $\mathcal{C}_0$ draws each R_i from $\mathsf{Enc}(m|\mathsf{L})$, whereas $\mathcal{C}_2$ draws R_i from a distribution which is independent of m and L; call this distribution $\mathcal{D}_{\mathsf{hid}}$. Moreover, $\mathcal{A}$ cannot distinguish between his interaction with $\mathcal{C}_0$ and $\mathcal{C}_1$ by the hiding of Com. Likewise, he cannot distinguish between his interaction with $\mathcal{C}_1$ and $\mathcal{C}_2$ by semantic security. □

Lemma 1. $(\mathsf{Enc}, \mathsf{Dec})$ *is computationally conditionally augmented non-malleable against*

$$\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}} = \big\{(f, g) \in \mathcal{F}_{\mathsf{split}} : f \text{ and } g \text{ polytime}\big\}.$$

Lemma 1 is proven in full in Section 6. In the remainder of this section we give a high-level overview.

Proof Idea. Fix $(f,g) \in \mathcal{F}_{\mathsf{split}}^{\mathsf{poly}}$. We describe a family of simulators for $\{\mathsf{V}^{\mathsf{L}}_{m,f,g}\}_{m,\mathsf{L}}$. $\mathsf{S}^{\mathsf{L}}_{f,g}$ will be one of two distributions depending on L. The first simply draws $\mathsf{R} \leftarrow \mathcal{D}_{\mathsf{hid}}$ and outputs $(\mathsf{R}, \perp_{\mathsf{com}})$. This will simulate $\{\mathsf{V}^{\mathsf{L}}_{m,f,g}\}_m$ whenever $\mathsf{p}_{\mathsf{L},m}$ is small for all $m \in \mathcal{M}$ where $\mathsf{p}_{\mathsf{L},m}$ is shorthand for $\Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\big(\tilde{m} \neq \perp_{\mathsf{com}}\big)$. In other words, if (f,g) always tampers to $\perp_{\mathsf{com}}$, then non-malleability follows from right state simulatability.

On the other hand, if $\mathsf{p}_{\mathsf{L},m}$ is large for all $m \in \mathcal{M}$ then $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ is identical to $\mathsf{V}^{\mathsf{L}_0}_{m,f_0,g_0}$, a conditional augmented tampering distribution of $(\mathsf{Enc}_0, \mathsf{Dec}_0)$, for some tampering functions $(f_0, g_0) \in \mathcal{F}_{\mathsf{split}}$ which are related to (f,g). In this case, we can use $\mathsf{S}^{\mathsf{L}_0}_{f_0,g_0}$ to construct $\mathsf{S}^{\mathsf{L}}_{f,g}$.

The final piece of the proof involves ruling out selective bot attacks. We show that if the likelihood of (f,g) tampering to $\perp_{\mathsf{com}}$ depends on the encoded message (*i.e.*, if $\mathsf{p}_{\mathsf{L},m}$ is significantly larger than $\mathsf{p}_{\mathsf{L},m'}$ for some m, m'), then (f,g) can be used to win the above hiding game with non-negligible advantage. Since f and g are polytime, either $\mathsf{p}_{\mathsf{L},m}$ is small for all m, or $\mathsf{p}_{\mathsf{L},m}$ is large for all m and so we are in one of the cases above.

The first point (when $\mathsf{p}_{\mathsf{L},m}$ is small for all $m \in \mathcal{M}$) is relatively straightforward, but we expand a bit on the second and third points. The functions (f_0, g_0) are defined as follows.

- **Random Choices:** Draw $k \leftarrow \mathsf{G}(1^\lambda)$, $\sigma \leftarrow \$$, $c^\$ \leftarrow \mathsf{E}_k(0)$, set $z = \mathrm{Com}(k,\sigma)$, and $(\cdot, \tilde{z}^\$) = g(c^\$, z)$.
- $f_0(\mathsf{L}_0)$**:** Compute $\big(\tilde{\mathsf{L}}_0, (\tilde{k}, \tilde{\sigma})\big) = f\big(\mathsf{L}_0, (k,\sigma)\big)$. Output $\tilde{\mathsf{L}}_0$.
- $g_0(\mathsf{R}_0)$**:** Draw $c \leftarrow \mathsf{E}_k(\mathsf{R}_0)$ and set $(\tilde{c}, \tilde{z}) = g(c,z)$. If $\tilde{z} \neq \tilde{z}^\$$, output $\perp_{\mathsf{com}}$. Otherwise, use superpolynomial time to break open $\tilde{z}$ and recover the pair $(\tilde{k}', \tilde{\sigma}')$. Output $\tilde{\mathsf{R}}_0 = \mathsf{D}_{\tilde{k}'}(\tilde{c})$.

Remark. Note that g_0 above does not run in polynomial time. It is possible to change the construction and get the proof to work using polynomial time (f_0, g_0), however the proof would become longer and more difficult.

The main observation is that whenever $(\mathsf{L}, c^\$)$ are such that $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k}, \tilde{\sigma})$, the distributions $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ and $\mathsf{V}^{\mathsf{L}_0}_{m,f_0,g_0}$ are the same (up to encrypting the right state output by $\mathsf{V}^{\mathsf{L}_0}_{m,f_0,g_0}$). In this case, we can simulate $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ using $\mathsf{S}^{\mathsf{L}_0}_{f_0,g_0}$, so it suffices to show that $c^\$$ exists such that $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k}, \tilde{\sigma})$. This follows from the semantic security of $(\mathsf{G}, \mathsf{E}, \mathsf{D})$ as $(c^\$, z)$ and $\mathsf{R} = (c, z) \leftarrow \mathsf{Enc}(m|\mathsf{L})$ differ only in their encrypted values.

We prove the claim of the third point above, that $\mathsf{p}_{\mathsf{L},m}$ and $\mathsf{p}_{\mathsf{L},m'}$ cannot differ very much, as it exhibbits a type of argument which will be very useful to us in the rest of the paper.

The Hiding Machine. We mention here one technique which will be very useful to us moving forward. The cryptography we have inserted into the right side of our code gives us a way to rule out certain tampering behavior via reductions to computational security. We refer to such arguments as "the hiding machine". At its core, the hiding machine is just a reduction to one of the hiding games above but there are many moving parts and so things often get quite complicated. Claim 4 below is a particularly simple example of the hiding machine in action and makes for a good first encounter. In general, the hiding machine is useful in ruling out various selective bot attacks.

Claim 4. *For any $m, m' \in \mathcal{M}$, $(f,g) \in \mathcal{F}_{\mathsf{split}}^{\mathsf{poly}}$ and non-negligible $\xi > 0$ we have* $\Pr_{\mathsf{L}}\Big[\big|\mathsf{p}_{\mathsf{L},m} - \mathsf{p}_{\mathsf{L},m'}\big| > \xi\Big] = \mathsf{negl}$.

Proof. Fix $m, m' \in \mathcal{M}$, $(f,g) \in \mathcal{F}_{\mathsf{split}}^{\mathsf{poly}}$ and non-negligible $\xi = \xi(\lambda) > 0$. Let BAD be the set of $\mathsf{L} \in \mathcal{L}$ such that $\mathsf{p}_{\mathsf{L},m} > \xi + \mathsf{p}_{\mathsf{L},m'}$, and suppose for contradiction that there is a non-negligible $\xi' = \xi'(\lambda) > 0$ such that $\Pr_{\mathsf{L}}\big[\mathsf{L} \in \mathsf{BAD}\big] \geq \xi'$. Fix $N = 3/(\xi\xi')$, $N' = \Omega(\lambda\xi^{-2})$ and consider the PPT adversary $\mathcal{A}$ who plays the $(N, N')-$way hiding game for $(\mathsf{Enc}, \mathsf{Dec})$ against a challenger $\mathcal{C}$ as follows.

- $\mathcal{A}$ sends m, m' to $\mathcal{C}$ and receives $\big(\mathsf{R}, \{\mathsf{R}_1\}, \ldots, \{\mathsf{R}_N\}\big)$.
- $\mathcal{A}$ computes $(\tilde{c}, \tilde{z}) = g(\mathsf{R})$ and $(\tilde{c}_i, \tilde{z}_i) = g(\mathsf{R}_i)$ for each $i = 1, \ldots, N$ and $\mathsf{R}_i \in \{\mathsf{R}_i\}$.

- For $i = 1, \ldots, N$, $\mathcal{A}$ sets $p_i = \Pr_{\mathsf{R}_i \in \{\mathsf{R}_i\}}\big(\tilde{z}_i = \tilde{z}\big)$ and outputs i^* such that p_{i^*} is maximal.

Note that if the random secret $\mathsf{L} \in \mathcal{L}$ chosen by $\mathcal{C}$ is in BAD then $\mathrm{Decom}(\tilde{z}) = (\tilde{k}, \tilde{\sigma})$ with probability at least ξ, where $f(\mathsf{L}) = \big(\tilde{\mathsf{L}}_0, (\tilde{k}, \tilde{\sigma})\big)$. If $\mathrm{Decom}(\tilde{z}) = (\tilde{k}, \tilde{\sigma})$, then for each i and $\mathsf{R}_i \in \{\mathsf{R}_i\}$, $\mathsf{Dec}(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}_i) \neq \perp_{\mathsf{com}}$ if and only if $\tilde{z}_i = \tilde{z}$. In this case, p_i approximates $\mathsf{p}_{\mathsf{L},m_i}$ where $m_i = m'$ if $i \neq i^*$ and $m_{i^*} = m$. Therefore,

$$p_{i^*} \geq \mathsf{p}_{\mathsf{L},m} - \frac{\xi}{3} > \mathsf{p}_{\mathsf{L},m'} + \xi - \frac{\xi}{3} \geq p_i + \frac{\xi}{3},$$

for all $i \neq i^*$. We have used the Chernoff-Hoeffding bound, facilitated by our choice of large N'. So we see that

$$\Pr\big(\mathcal{A} \text{ wins}\big) \geq \Pr\big(\mathsf{L} \in \mathsf{BAD}\big)\Pr\big(\mathrm{Decom}(\tilde{z}) = (\tilde{k}, \tilde{\sigma})\big|\mathsf{L} \in \mathsf{BAD}\big)(1 - \mathsf{negl}) = \xi\xi' - \mathsf{negl} > \frac{2}{N},$$

and so $\mathcal{A}$ breaks the right state simulatability of $(\mathsf{Enc}, \mathsf{Dec})$. □

4 The Basic Protocol

The protocol is shown in Figure 2.

Setup: Let Com be a non-interactive, perfectly binding commitment scheme. Let $(\mathsf{Enc}, \mathsf{Dec})$ be a computational, conditional, augmented non-malleable code. Fix a large prime q. Let $\mathsf{id} \in \{0,1\}^\lambda$ be $\mathcal{C}$'s identity.

Commiter's Private Input: $v \in \mathcal{M}_{\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}}$ to be committed to.

Commit Phase:

1. $\mathcal{C} \to \mathcal{R}$**:** Set $m = v \circ \mathsf{id}$ and draw $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m)$, where $\mathsf{L} \in \mathcal{L} \subset \mathbb{Z}_q$. Choose random $r \in \mathbb{Z}_q$ and send $\mathrm{Com}(\mathsf{L} \circ r)$ to $\mathcal{R}$.
2. $\mathcal{R} \to \mathcal{C}$**:** Send random challenge $\alpha \in \mathbb{Z}_q^*$.
3. $\mathcal{C} \to \mathcal{R}$**:** Send response $a = r\alpha + \mathsf{L} \in \mathbb{Z}_q$ and also send R.

Decommit Phase:

1. $\mathcal{C} \to \mathcal{R}$**:** Open the commitment sent in step 1. Let $\mathsf{L}' \circ r' \in \mathbb{Z}_q$ be the decommitted value.

Receiver's Output: If L' and r' do not satisfy $r'\alpha + \mathsf{L}' = a$ then output the special symbol $\perp_{\mathsf{inc}}$. Otherwise, compute $m' = \mathsf{Dec}(\mathsf{L}', \mathsf{R})$ and parse $m' = v' \circ \mathsf{id}'$. Output v' if $\mathsf{id}' = \mathsf{id}$, $\perp_{\mathsf{id}}$ if not.

Figure 2: Non-malleable commitment scheme $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$.

Claim 5. *$\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ is a perfectly binding commitment scheme.*

Proof Sketch. Perfect binding follows immediately from the perfect binding of Com. Computational hiding follows in a straightforward fashion from the hiding of Com and the well known fact that any split-state non-malleable code is also a 2-out-of-2 secret sharing scheme. □

Theorem 1. *$\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ is non-malleable against a synchronizing adversary.*

A Hiding Game For $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$. Before proving Theorem 1, we specify a hiding game for $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$, analogous to the hiding game for $(\mathsf{Enc}, \mathsf{Dec})$. Consider the following interaction between a challenger $\mathcal{C}$ and a PPT adversary $\mathcal{A}$.

- **Partial Transcript:** $\mathcal{C}$ chooses $\mathsf{L} \leftarrow \mathcal{L}$, $r \leftarrow \mathbb{Z}_q$ and sends $\mathbf{Com}_{\mathsf{L}} = \big(\mathrm{Com}(\mathsf{L}), \mathrm{Com}(r)\big)$ to $\mathcal{A}$, $\mathcal{A}$ returns $\alpha \in \mathbb{Z}_q$ and receives $a = r\alpha + \mathsf{L}$ from $\mathcal{C}$.
- **Message Choice:** $\mathcal{A}$ chooses $m, m' \in \mathcal{M}$ and sends them to $\mathcal{C}$.
- **Challenge Message:** $\mathcal{C}$ chooses $b \leftarrow \{0,1\}$ and sends the pair $(\mathsf{R}_0, \mathsf{R}_1)$ to $\mathcal{A}$ where $\mathsf{R}_b \leftarrow \mathsf{Enc}(m|\mathsf{L})$ and $\mathsf{R}_{1-b} \leftarrow \mathsf{Enc}(m'|\mathsf{L})$.
- **Guess:** $\mathcal{A}$ outputs a guess $b' \in \{0,1\}$ and wins if $b' = b$.

Just as for the hiding game of the code $(\mathsf{Enc}, \mathsf{Dec})$, we will usually use an $(N, N')-$way variant of the above game, where the challenge message is $\big(\mathsf{R}, \{\mathsf{R}_1\}, \ldots, \{\mathsf{R}_N\}\big)$ where $\mathsf{R} \leftarrow \mathsf{Enc}(m|\mathsf{L})$, each $^{\#}\{\mathsf{R}_i\} = N'$, and also $\mathsf{R}_i \leftarrow \mathsf{Enc}(m'|\mathsf{L})$ for all $\mathsf{R}_i \in \{\mathsf{R}_i\}$ and all i except for a random i^*, for which $\mathsf{R}_{i^*} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ for all $\mathsf{R}_{i^*} \in \{\mathsf{R}_{i^*}\}$. In the $(N, N')-$way variant, $\mathcal{A}$ wins if he guesses i^*.

Claim 6. *If* Com *is computationally hiding and* $(\mathsf{Enc}, \mathsf{Dec})$ *has the hiding property then for all PPT adversaries $\mathcal{A}$, the probability that $\mathcal{A}$ wins the above game (resp. its $(N, N')-$way variant) is at most* $1/2 + \mathsf{negl}$ *(resp.* $1/N + \mathsf{negl}$*).*

5 Proof of Non-Malleability (Theorem 1)

5.1 Notation

Transcripts. Suppose a PPT man-in-the-middle, M, participates in two protocol executions. We denote the transcript of M's view with the letter $\mathbb{T}$. So

$$\mathbb{T} = \big(\mathsf{id}, \tilde{\mathsf{id}}, \mathrm{Com}(\mathsf{L}), \mathrm{Com}(r), \mathrm{Com}(\tilde{\mathsf{L}}), \mathrm{Com}(\tilde{r}), \tilde{\alpha}, \alpha, a, \mathsf{R}, \tilde{a}, \tilde{\mathsf{R}}\big).$$

We write $\mathbf{Com}_{\mathsf{L}}$ for $\mathrm{Com}(\mathsf{L})$ and $\mathrm{Com}(r)$. Note $\mathbf{Com}_{\mathsf{L}}$ specifies a linear polynomial $\varphi(x) = rx + \mathsf{L}$ which $\mathcal{C}$ uses to answer M's query α. We will usually write a transcript $\mathbb{T}$ more consicely as $\mathbb{T} = \big(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a, \mathsf{R}\big)$, surpressing $\mathcal{C}$'s identity id and the quantities which are outputs of M. Since without loss of generality M is deterministic, these values uniquely define a full transcript.

Partial Transcripts. We also will find it useful to speak of partial transcripts, as this will let us isolate certain random choices made during the execution of $\mathbb{T}$. We use τ to denote the partial transcript where $\mathcal{C}$'s value R remains unspecified. We write τ consicely as $\tau = (\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a)$. Note that M's third message is not specified given τ, however τ extends to a full transcript $\mathbb{T}$ once R is chosen. We write this full transcript $\mathbb{T}(\tau, \mathsf{R})$.

The Distribution M_m^τ and Distinguisher D^τ. Our goal is to use a MIM who breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ to violate the security of the code $(\mathsf{Enc}, \mathsf{Dec})$. In order to do this we make some notational changes which syntactically relate M to the code's non-malleability game. By definition, if M breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ then there exists $v \in \mathcal{M}_{\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}}$, a PPT distinguisher D, and non-negligible $\delta = \delta(\lambda) > 0$ such that

$$\Big| \Pr_{(\mathbb{T}, \tilde{v}) \leftarrow \mathbf{MIM}_v} \big(\mathrm{D}(\mathbb{T}, \tilde{v}) = 1\big) - \Pr_{(\mathbb{T}, \tilde{v}) \leftarrow \mathbf{MIM}_0} \big(\mathrm{D}(\mathbb{T}, \tilde{v}) = 1\big) \Big| = \delta. \tag{1}$$

Let $m = v \circ \mathsf{id}$, $m' = 0 \circ \mathsf{id}$ and $\tilde{m} = \tilde{v} \circ \tilde{\mathsf{id}}$. So $m, m' \in \mathcal{M}$ are the messages encoded during the left executions of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ in the real/ideal world, and $\tilde{m} \in \mathcal{M}$ is the message encoded on the right. For a given partial transcript

$\tau = \left(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a\right)$, let M^{τ}_{m} be the distribution which draws $\mathsf{R} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ and outputs $(\mathsf{R}, \tilde{m})$, where $\tilde{m}$ is M's encoded message in $\mathbb{T}(\tau, \mathsf{R})$. Let D^{τ} be the PPT distinguisher which on input $(\mathsf{R}, \tilde{m})$, sets $\mathbb{T} = \mathbb{T}(\tau, \mathsf{R})$, parses $\tilde{m} = \tilde{v} \circ \tilde{\mathsf{id}}'$ and outputs $\mathrm{D}(\mathbb{T}, \tilde{v})$. With these notational changes in place, (1) gives

$$\Pr_{\tau}\left[\left|\Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^{\tau}_{m}}\left(\mathrm{D}^{\tau}\left(\mathsf{R},\tilde{m}\right)=1\right) - \Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^{\tau}_{m'}}\left(\mathrm{D}^{\tau}\left(\mathsf{R},\tilde{m}\right)=1\right)\right| \geq \frac{\delta}{2}\right] \geq \frac{\delta}{2}. \tag{2}$$

Note that since M is required to produce a commitment using a tag $\tilde{\mathsf{id}} \neq \mathsf{id}$, when $(\mathsf{R}, \tilde{m})$ is drawn from M^{τ}_{m} or $\mathsf{M}^{\tau}_{m'}$, we will always have $\tilde{m} \notin \{m, m'\}$.

Definition 12 (Malleable Partial Transcripts). *For $m, m' \in \mathcal{M}$, write $\tau \in \mathsf{MAUL}_{m,m'}$ if*

$$\left|\Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^{\tau}_{m}}\left(\mathrm{D}^{\tau}\left(\mathsf{R},\tilde{m}\right)=1\right) - \Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^{\tau}_{m'}}\left(\mathrm{D}^{\tau}\left(\mathsf{R},\tilde{m}\right)=1\right)\right| \geq \delta/2.$$

So if M breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$, there exist $m, m' \in \mathcal{M}$ such that $\Pr_{\tau}\left[\tau \in \mathsf{MAUL}_{m,m'}\right] \geq \delta/2$.

5.2 Proof Overview

At this point, we go through the proof at a high level, in order to highlight the key ideas. As mentioned in the intro, our approach is to use an M who mauls $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ to construct polynomial time split state tampering functions $(f, g) \in \mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$ which maul the code $(\mathsf{Enc}, \mathsf{Dec})$. In fact, we will construct an efficiently samplable distribution $\mathcal{D}_{\mathsf{split}}$ on $\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$ such that with non-negligible probability over $(f, g) \leftarrow \mathcal{D}_{\mathsf{split}}$, (f, g) mauls $(\mathsf{Enc}, \mathsf{Dec})$. We proceed in two steps. First, we use M to define an efficiently samplable distribution $\mathcal{D}_{\mathrm{M}}$ which outputs polytime (f, g) which maul $(\mathsf{Enc}, \mathsf{Dec})$ but are not split-state. Then we show that $\mathcal{D}_{\mathrm{M}}$ is indistinguishable from a distribution $\mathcal{D}_{\mathsf{split}}$ on $\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$.

The functions (f, g) output by $\mathcal{D}_{\mathrm{M}}$ share a partial transcript $\tau = \left(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a\right)$ and $\mathsf{R}_{\$} \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$ for some arbitrary fixed $m^* \in \mathcal{M}$. This defines a full transcript $\mathbb{T} = \mathbb{T}(\tau, \mathsf{R}_{\$})$, let $\tilde{a}_{\$}$ be M's response in the third message of $\mathbb{T}$. Given L, f extracts a candidate $\tilde{\mathsf{L}}$ from M by rewinding M and asking a new challenge $\tilde{\beta}$, answering with $(b, \mathsf{R}_{\$})$ on the left, where b is computed using L and the point (α, a) from τ. Defining $g(\mathsf{R})$ is simpler: it sends (a, R) to M receiving $(\tilde{a}, \tilde{\mathsf{R}})$, if $\tilde{a} = \tilde{a}_{\$}$ it outputs $\tilde{\mathsf{R}}$, if not it outputs $\perp_{\mathsf{inc}}$.

Clearly, (f, g) are not split-state as the randomness $(\tau, \mathsf{R}_{\$}) = \left(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a, \mathsf{R}_{\$}\right)$ shared by both functions depends on L. However, $(\tau, \mathsf{R}_{\$})$ only depends on information which computationally hides L, and we show in Section 5.5 that indeed $\mathcal{D}_{\mathrm{M}} \approx_{\mathsf{c}} \mathcal{D}_{\mathsf{split}}$ where (f, g) output by $\mathcal{D}_{\mathsf{split}}$ are defined using bogus randomness $(\tau, \mathsf{R}_{\$})$ drawn from some distribution independent of L.

It is much harder to show that with non-negligible probability, $(f, g) \leftarrow \mathcal{D}_{\mathrm{M}}$ mauls $(\mathsf{Enc}, \mathsf{Dec})$. There are two main issues, which correspond exactly to the two additional properties we need from our non-malleable code. The first is that the tampered value output by (f, g) can only be equal to M's committed value if M's response $\tilde{a}_{\$}$ in $\mathbb{T}(\tau, \mathsf{R}_{\$})$ is correct. For this, we use the right state simulatability of $(\mathsf{Enc}, \mathsf{Dec})$. Since the only difference between $\mathsf{R}_{\$}$ and the real right state R is that $(\mathsf{L}, \mathsf{R}_{\$})$ is an encoding of m^* while (L, R) is an encoding of m, we can show that $\tilde{a}_{\$}$ is correct with roughly the same probability that $\tilde{a}$ is. This portion of our proof is in Section 5.3. We then find ourselves in one of two cases. Either the probability of correctness is large for both, or it is small for both in which case M is always committing to $\perp_{\mathsf{inc}}$. In the second case, he is not mauling. In the first case we show that (f, g) output M's committed value with non-negligible probability that is independent of whether M is mauling or not. It follows that if M mauls $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ with non-negligible probability, then (f, g) mauls $(\mathsf{Enc}, \mathsf{Dec})$ with related non-negligible probability. This part of our proof is in Section 5.4.

The second issue is that (f, g) are defined using τ which contains L. This means that the mauling experiment for (f, g) given a message m is conditional: draw $\mathsf{R} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ where L is as in τ, compute

$(\tilde{\mathsf{L}}, \tilde{\mathsf{R}}) = \big(f(\mathsf{L}), g(\mathsf{R})\big)$ and output $\tilde{m} = \mathsf{Dec}(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$. This is in contrast to the usual security game for non-malleable codes where (f, g) are fixed and then $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m)$ are both drawn during the tampering experiment, and it explains our need for starting with $(\mathsf{Enc}, \mathsf{Dec})$ which are conditionally augmented non-malleable.

5.3 Ruling out Selective $\perp_{\mathsf{inc}}$ Attacks

Recall that $\tilde{v} = \perp_{\mathsf{inc}}$ when M's response $\tilde{a}$ is incorrect. In this section we use the shorthand

$$\mathsf{p}_{\tau,m} = \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{M}^{\tau}_{m}}\big(\tilde{v} \neq \perp_{\mathsf{inc}}\big)$$

and we will prove that if M is mauling then he is doing so by answering correctly. The main lemma of this section is the following.

Lemma 2. *Suppose* M *breaks the non-malleability of* $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ *with non-negligible probability* δ*. Then there exist* $m, m' \in \mathcal{M}$ *and non-negligible* $\delta' > 0$ *such that for all* $m^* \in \mathcal{M}$

$$\Pr_{\tau}\left[\tau \in \mathsf{MAUL}_{m,m'} \;\&\; \mathsf{p}_{\tau,m^*} \geq \delta'\right] \geq \frac{\delta}{4} - \mathsf{negl}.$$

Lemma 2 follows immediately from (2) and Claims 7 and 8 which rule out two separate types of mauling behavior.

Claim 7. *For all* $m, m' \in \mathcal{M}$ *and non-negligible* $\xi > 0$ *we have:*

$$\Pr_{\tau}\left[\Big|\mathsf{p}_{\tau,m} - \mathsf{p}_{\tau,m'}\Big| > \xi\right] = \mathsf{negl}.$$

Proof. We utilize the hiding machine. Fix $m, m' \in \mathcal{M}$ and non-negligible $\xi > 0$. Let BAD be the set of partial transcripts τ for which $\mathsf{p}_{\tau,m} > \xi + \mathsf{p}_{\tau,m'}$ and suppose for contradiction that there is some non-negligible $\xi' > 0$ such that $\Pr\big(\tau \in \mathsf{BAD}\big) \geq \xi'$. Set $N = 4/(\xi\xi')$, $N' = \Omega\big(\lambda\xi^{-2}\big)$ and consider the PPT adversary $\mathcal{A}$ who interacts with $\mathcal{C}$ in the $(N, N')-$way hiding game for $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ as follows.

- $\mathcal{A}$ instantiates M. Upon receiving $\mathbf{Com}_{\mathsf{L}}$ from $\mathcal{C}$, it plays the first two rounds of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ with M, giving input $\mathbf{Com}_{\mathsf{L}}$ and uniform $\tilde{\alpha}$ and receiving α in the second round of the left interaction. $\mathcal{A}$ sends α to $\mathcal{C}$ and receives a. This defines a partial transcript $\tau = \big(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a\big)$.
- $\mathcal{A}$ sends (m, m') to $\mathcal{C}$ and receives challenge $\big(\mathsf{R}, \{\mathsf{R}_1\}, \ldots, \{\mathsf{R}_N\}\big)$. $\mathcal{A}$ forwards (a, R) to M and receives $(\tilde{a}, \tilde{\mathsf{R}})$. This defines a full transcript $\mathbb{T} = \mathbb{T}(\tau, \mathsf{R})$. Moreover, for all $i = 1, \ldots, N$ and $\mathsf{R}_i \in \{\mathsf{R}_i\}$, $\mathcal{A}$ sends (a, R_i) to M and receives $(\tilde{a}_i, \tilde{\mathsf{R}}_i)$, defining transcripts $\{\mathbb{T}_i\}$ for $i = 1, \ldots, N$, where $\mathbb{T}_i = \mathbb{T}(\tau, \mathsf{R}_i)$.
- $\mathcal{A}$ computes $p_i = \Pr_{\mathsf{R}_i \in \{\mathsf{R}_i\}}\big(\tilde{a}_i = \tilde{a}\big)$, and outputs i^* such that p_{i^*} is maximal.

Note that if $\tau \in \mathsf{BAD}$, then M's response $\tilde{a}$ in $\mathbb{T}$ is correct with probability at least ξ. Moreover, if $\tilde{a}$ is correct then $\tilde{v} \neq \perp_{\mathsf{inc}}$ in $\mathbb{T}_i$ if and only if $\tilde{a}_i = \tilde{a}$. Therefore, conditioned on $\tau \in \mathsf{BAD}$ and $\tilde{a}$ being correct, we see that

$$p_{i^*} \geq \mathsf{p}_{\tau,m} - \frac{\xi}{3} > \mathsf{p}_{\tau,m'} + \xi - \frac{\xi}{3} \geq p_i + \frac{\xi}{3},$$

for all $i \neq i^*$ with probability at least $1 - 2^{-\Omega(\lambda)}$. We have used the Chernoff-Hoeffding bound, made possible by our choice of large enough N'. We conclude that

$$\Pr\big(\mathcal{A} \text{ wins}\big) \geq \xi\xi'(1 - \mathsf{negl}) > \frac{2}{N},$$

which violates the security of the hiding game for $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$. □

Claim 8. *Let $\delta = \delta(\lambda) > 0$ be as the statement of Lemma 2. For all $m, m' \in \mathcal{M}$, we have:*

$$\Pr_\tau\left[\tau \in \mathsf{MAUL}_{m,m'} \;\&\; \mathsf{p}_{\tau,m} < \lambda^{-2}\delta^3\right] \leq \frac{\delta}{4}.$$

Proof. This is another invocation of the hiding machine. Fix $m, m' \in \mathcal{M}$, let BAD' be the partial transcripts $\tau \in \mathsf{MAUL}_{m,m'}$ such that $\mathsf{p}_{\tau,m} < \lambda^{-2}\delta^3$, and suppose for contradiction that $\Pr_\tau\big[\tau \in \mathsf{BAD}'\big] > \delta/4$. For almost all $\tau \in \mathsf{BAD}'$ (in particular, for all $\tau \in \mathsf{BAD}'$ except for those for which Claim 7 does not hold), we have:

$$\begin{aligned}
\frac{\delta}{2} &\leq \left|\Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^\tau_m}\Big(\mathrm{D}^\tau(\mathsf{R},\tilde{m}) = 1\Big) - \Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^\tau_{m'}}\Big(\mathrm{D}^\tau(\mathsf{R},\tilde{m}) = 1\Big)\right| \\
&\leq \left|\Pr_{\mathsf{M}^\tau_m}\Big(\mathrm{D}^\tau(\mathsf{R},\tilde{m}) = 1 \;\&\; \tilde{v} = \perp_{\mathsf{inc}}\Big) - \Pr_{\mathsf{M}^\tau_{m'}}\Big(\mathrm{D}^\tau(\mathsf{R},\tilde{m}) = 1 \;\&\; \tilde{v} = \perp_{\mathsf{inc}}\Big)\right| + \mathsf{p}_{\tau,m} + \mathsf{p}_{\tau,m'} \\
&< \left|\Pr_{\mathsf{M}^\tau_m}\Big(\mathrm{D}^\tau(\mathsf{R},\perp_{\mathsf{inc}}) = 1\Big|\tilde{v} = \perp_{\mathsf{inc}}\Big) - \Pr_{\mathsf{M}^\tau_{m'}}\Big(\mathrm{D}^\tau(\mathsf{R},\perp_{\mathsf{inc}}) = 1\Big|\tilde{v} = \perp_{\mathsf{inc}}\Big)\right| + 3\lambda^{-2}\delta^3 + \mathsf{negl}.
\end{aligned}$$

Now, set $N = 17/\delta$, $N' = \Omega(\lambda\delta^{-2})$ and consider the $\mathcal{A}$ who plays the hiding game against $\mathcal{C}$ as follows.

- $\mathcal{A}$ instantiates M and obtains partial transcript τ. $\mathcal{A}$ sends m, m', receives $\big(\mathsf{R}, \{\mathsf{R}_1\}, \ldots, \{\mathsf{R}_N\}\big)$, and sets $\mathbb{T}_i = \mathbb{T}(\tau, \mathsf{R}_i)$, for all $\mathsf{R}_i \in \{\mathsf{R}_i\}$.
- $\mathcal{A}$ computes $p_i = \Pr_{\mathsf{R}_i\in\{\mathsf{R}_i\}}\big(\mathrm{D}^\tau(\mathsf{R}_i, \perp_{\mathsf{inc}}) = 1\big)$ and outputs i^* such that p_{i^*} is maximal.

If $\tau \in \mathsf{BAD}'$ then with probability at least 1/2, M will answer incorrectly in every $\mathbb{T}_i \in \{\mathbb{T}_i\}$ for all i, and so M's commitment in every $\mathbb{T}_i$ is to $\perp_{\mathsf{inc}}$. Conditioned on every commitment being to $\perp_{\mathsf{inc}}$, p_i approximates $\Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^\tau_{m_i}}\big(\mathrm{D}^\tau(\mathsf{R},\perp_{\mathsf{inc}}) = 1\big|\tilde{v} = \perp_{\mathsf{inc}}\big)$ where $m_{i^*} = m$ and $m_i = m'$ when $i \neq i^*$. In this case, we have by Chernoff-Hoeffding,

$$p_{i^*} \geq \Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^\tau_m}\Big(\mathrm{D}^\tau(\mathsf{R},\perp_{\mathsf{inc}}) = 1\Big|\tilde{v} = \perp_{\mathsf{inc}}\Big) - \frac{\delta}{9} > \Pr_{(\mathsf{R},\tilde{m})\leftarrow\mathsf{M}^\tau_{m'}}\Big(\mathrm{D}^\tau(\mathsf{R},\perp_{\mathsf{inc}}) = 1\Big|\tilde{v} = \perp_{\mathsf{inc}}\Big) + \frac{2\delta}{9} \geq p_i + \frac{\delta}{9},$$

for all $i \neq i^*$ with probability $1 - 2^{-\Omega(\lambda)}$. And so $\Pr\big(\mathcal{A}\text{ wins}\big) \geq \frac{\delta}{8} - \mathsf{negl} > \frac{2}{N}$, which violates the security of the hiding game for $\langle\mathcal{C},\mathcal{R}\rangle_{\mathsf{synch}}$. □

5.4 The Distribution $\mathcal{D}_{\mathrm{M}}$

We now use M to define a polynomial time sampleable distribution, $\mathcal{D}_{\mathrm{M}}$, which outputs a tampering function pair (f, g), as follows.

- **Random Choices:** Instantiate M and play the first two rounds of $\langle\mathcal{C},\mathcal{R}\rangle_{\mathsf{synch}}$, obtaining a partial transcript $\tau = \big(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a\big)$ where $a = \varphi(\alpha)$ and $\varphi(x)$ is the linear map specified by $\mathbf{Com}_{\mathsf{L}}$. Draw $\mathsf{R}_\$ \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$ for some arbitrary fixed $m^* \in \mathcal{M}$ and let $\tilde{a}_\$$ be M's response in the full transcript $\mathbb{T}(\tau, \mathsf{R}_\$)$. Finally draw $\tilde{\beta} \leftarrow \mathbb{Z}_q$.
- $f_{\tau,\mathsf{R}_\$,\tilde{\beta}}(\mathsf{L})$**:** Let $\varphi(x)$ be the unique linear function with constant term L and $\varphi(\alpha) = a$.
  - rewind M back to the second message of the right interaction and ask $\tilde{\beta}$, receive β on the left;
  - send $(b, \mathsf{R}_\$)$ where $b = \varphi(\beta)$ and receive $(\tilde{b}, \cdot)$ on the right;
  - output $\tilde{\mathsf{L}}$, the constant term of the line spanned by $\big\{(\tilde{\alpha}, \tilde{a}_\$), (\tilde{\beta}, \tilde{b})\big\}$.
- $g_{\tau,\mathsf{R}_\$}(\mathsf{R})$**:** Let $(\tilde{a}, \tilde{\mathsf{R}})$ be M's final message in $\mathbb{T}(\tau, \mathsf{R})$. If $\tilde{a} = \tilde{a}_\$$ output $\tilde{\mathsf{R}}$, otherwise $\perp_{\mathsf{inc}}$.

- **Output:** $(f, g) = (f_{\tau,\mathsf{R}_\$,\tilde{\beta}}, g_{\tau,\mathsf{R}_\$})$.

Notice (f, g) output by $\mathcal{D}_{\mathrm{M}}$ are not split-state as the randomness $(\tau, \mathsf{R}_\$)$ shared by both f and g depends on L. Nonetheless, we show in Section 5.5 below that $\mathcal{D}_{\mathrm{M}} \approx_{\mathsf{c}} \mathcal{D}_{\mathsf{split}}$, a distribution on $\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$. Combined with the next lemma, this shows that $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ is non-malleable: an M which breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ can be used to construct a distribution $\mathcal{D}_{\mathsf{split}}$ on $\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$ which breaks the security of the code $(\mathsf{Enc}, \mathsf{Dec})$.

Lemma 3. *Let $\delta, \delta' > 0$ be as in the statement of Lemma 2. If M breaks the non-malleability of $\langle \mathcal{C}, \mathsf{R} \rangle$ then there exist $m, m' \in \mathcal{M}$ such that*

$$\Pr_{\mathsf{L},(f,g)}\left[\left|\Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\Big(\mathrm{D}^{\tau}(\mathsf{R},\tilde{m})=1\Big) - \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m',f,g}}\Big(\mathrm{D}^{\tau}(\mathsf{R},\tilde{m}))=1\Big)\right| > \frac{\delta}{2}\right] > \frac{(\delta\delta')^3}{256} - \mathsf{negl},$$

where the outer probability is over $\mathsf{L} \leftarrow \mathcal{L}$ and $(f, g) = (f_{\tau,\mathsf{R}_\$,\tilde{\beta}}, g_{\tau,\mathsf{R}_\$}) \leftarrow \mathcal{D}_{\mathrm{M}}$, where $\tau = (\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a)$

Proof Idea. The randomness needed in order to draw $(f, g) \leftarrow \mathcal{D}_{\mathrm{M}}$ consists of a random partial transcript $\tau = \big(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, a\big)$, $\mathsf{R}_\$ \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$ and $\tilde{\beta} \leftarrow \mathbb{Z}_q$. Given these choices, the distributions $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ and M^{τ}_{m} are very similar: both draw $\mathsf{R} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ and output $(\mathsf{R}, \tilde{m})$. We prove Lemma 3 by showing that whenver $\big(\tau, \mathsf{R}_\$, \tilde{\beta}\big)$ is such that M's response $\tilde{a}_\$$ is correct, $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ and M^{τ}_{m} are actually identical for all $m \in \mathcal{M}$. The proof then follows from the observation that $\tilde{a}_\$$ is correct with non-negligible probability, which uses Lemma 2 and the right state simulatability of $(\mathsf{Enc}, \mathsf{Dec})$.

Proof of Lemma 3. For randomness $(\tau, \mathsf{R}_\$) = \big(\mathbf{Com}_{\mathsf{L}}, \tilde{\alpha}, r\alpha + \mathsf{L}, \mathsf{R}_\$\big)$, say the "extraction event", denoted EXT, occurs whenever $\tilde{a}_\$$ is correct in $\mathbb{T}(\tau, \mathsf{R}_\$)$, and

$$\Pr_{\tilde{\beta}}\Big(\tilde{b} \text{ correct in } \mathbb{T}\big(\mathbf{Com}_{\mathsf{L}}, \tilde{\beta}, r\beta + \mathsf{L}, \mathsf{R}_\$\big)\Big) \geq \frac{(\delta\delta')^2}{32}.$$

If M mauls $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ then there exist messages $m, m' \in \mathcal{M}$ such that

$$\Pr_{\tau,\mathsf{R}_\$}\big(\tau \in \mathsf{MAUL}_{m,m'} \;\&\; \mathsf{EXT}\big) \geq \frac{\delta\delta'}{4} - \frac{\delta\delta'}{8} - \mathsf{negl} = \frac{\delta\delta'}{8} - \mathsf{negl},$$

using Lemma 2 and Bayes' theorem. If $\tilde{a}_\$$ is correct then $g(\mathsf{R})$ identifies when M is committing to $\perp_{\mathsf{inc}}$ on the right ($\tilde{a}$ is correct if and only if $\tilde{a} = \tilde{a}_\$$), and so outputs the correct $\tilde{\mathsf{R}}$. Moreover, if $\tilde{a}_\$$ and $\tilde{b}$ are correct then $f(\mathsf{L})$ outputs the correct $\tilde{\mathsf{L}}$. So we see that when $\tau \in \mathsf{MAUL}_{m,m'}$ and EXT occurs and $\tilde{b}$ is correct (all of which happen with probability at least $(\delta\delta')^3/256 - \mathsf{negl}$), then $\mathsf{V}^{\mathsf{L}}_{m,f,g} \equiv \mathsf{M}^{\tau}_{m}$ for all $m \in \mathcal{M}$ and so since $\tau \in \mathsf{MAUL}_{m,m'}$,

$$\left|\Pr\big(\mathrm{D}^{\tau}(\mathsf{V}^{\mathsf{L}}_{m,f,g}) = 1\big) - \Pr\big(\mathrm{D}^{\tau}(\mathsf{V}^{\mathsf{L}}_{m',f,g}) = 1\big)\right| > \frac{\delta}{2}.$$

□

5.5 A Hybrid Argument to Prove $\mathcal{D}_{\mathrm{M}} \approx_{\mathsf{c}} \mathcal{D}_{\mathsf{split}}$

Note that the distribution $\mathcal{D}_{\mathrm{M}}$ from the previous section does not output split state (f, g) as the randomness $(\tau, \mathsf{R}_\$) = \big(\mathrm{Com}(\mathsf{L}), \mathrm{Com}(r), \tilde{\alpha}, a, \mathsf{R}_\$\big)$ shared between f and g depends in three ways on L: 1) τ contains a commitment to L, 2) $a = r\alpha + \mathsf{L}$, and 3) $\mathsf{R}_\$ \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$. We show in this section, however, that $\mathcal{D}_{\mathrm{M}}$ is computationally indistinguishable from a polynomial time sampleable distribution $\mathcal{D}_{\mathsf{split}}$ on $\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$. This, together with Lemma 3, completes the proof that $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ is non-malleable.

$\mathcal{D}_0 = \mathcal{D}_{\mathrm{M}} -$ This is the distribution defined above. It draws $\mathsf{L} \leftarrow \mathcal{L}$, a and also random partial transcript $\tau = \big(\mathrm{Com}(\mathsf{L}), \mathrm{Com}(r), \tilde{\alpha}, a\big)$ where $a = r\alpha + \mathsf{L}$, $\mathsf{R}_\$ \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$ and outputs $(f_0, g_0) = (f_{\tau,\mathsf{R}_\$}, g_{\tau,\mathsf{R}_\$})$.

$\mathcal{D}_1 -$ This distribution outputs functions (f_1, g_1) which behave exactly like (f_0, g_0) except that they are seeded with $(\tau, \mathsf{R}_\$) = \big(\mathrm{Com}(\mathsf{L}), \mathrm{Com}(r), \tilde{\alpha}, a, \mathsf{R}_\$\big)$, where $a \leftarrow \mathbb{Z}_q$ is random instead of equal to $r\alpha + \mathsf{L}$.

$\mathcal{D}_2 -$ This outputs (f_2, g_2) which, again, differ from (f_1, g_1) only in their shared randomness. This time $(\tau, \mathsf{R}_\$) = \big(\mathrm{Com}(0), \mathrm{Com}(r), \tilde{\alpha}, a, \mathsf{R}_\$\big)$, where $a \in \mathbb{Z}_q$ is random. Now the only dependence on L is that $\mathsf{R}_\$ \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$.

$\mathcal{D}_3 = \mathcal{D}_{\mathsf{split}} -$ This outputs (f_3, g_3) which are the same as (f_2, g_2) except that $\mathsf{R}_\$ \leftarrow \mathcal{D}_{\mathsf{hid}}$, the distribution on $\mathcal{R}$ guaranteed by the right state simulatability of $(\mathsf{Enc}, \mathsf{Dec})$. Since $(\tau, \mathsf{R}_\$)$ no longer depends on L, $(f_3, g_3) \in \mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$.

Claim 9. $\mathcal{D}_0 \approx_{\mathsf{c}} \mathcal{D}_1 \approx_{\mathsf{c}} \mathcal{D}_2 \approx_{\mathsf{c}} \mathcal{D}_3$.

Proof. The first two indistinguishabilities follow from the hiding of Com. For the first, consider an adversary $\mathcal{A}$ who interacts with a challenger $\mathcal{C}$ in the hiding game by choosing $r_0, r_1 \in \mathbb{Z}_q$ at random and sending (r_0, r_1) to $\mathcal{C}$, receiving a commitment $z = \mathrm{Com}(r_b)$ for a random $b \in \{0, 1\}$. Then $\mathcal{A}$ draws $\mathsf{L} \leftarrow \mathcal{L}$ and $\tilde{\alpha} \leftarrow \mathbb{Z}_q$ at random and sends $\big(\mathrm{Com}(\mathsf{L}), z\big)$ and $\tilde{\alpha}$ to M (corresponding to the first message of the left interaction and the second message of the right interaction), receiving α as the second message on the left. $\mathcal{A}$ sets $a = r_0\alpha + \mathsf{L}$, draws $\mathsf{R}_\$ \leftarrow \mathsf{Enc}(m_\$|\mathsf{L})$ and outputs $\big(\mathrm{Com}(\mathsf{L}), z, \tilde{\alpha}, a, \mathsf{R}_\$\big)$. If $b = 0$ then $\mathcal{A}$'s output is distributed according to $\mathcal{D}_0$, while if $b = 1$, $\mathcal{A}$'s output is distributed like $\mathcal{D}_1$. This proves that $\mathcal{D}_0 \approx_{\mathsf{c}} \mathcal{D}_1$; $\mathcal{D}_1 \approx_{\mathsf{c}} \mathcal{D}_2$ follows even more readily. Finally, $\mathcal{D}_2 \approx_{\mathsf{c}} \mathcal{D}_3$ follows from the hiding property of $(\mathsf{Enc}, \mathsf{Dec})$. □

It follows from Lemma 3 and Claim 9 that if M breaks the non-malleability of $\langle \mathcal{C}, \mathsf{R} \rangle$ then there exist $m, m' \in \mathcal{M}$ and non-negligible $\delta, \delta' > 0$ such that

$$\Pr_{\mathsf{L},(f,g)}\left[\left|\Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\Big(\mathrm{D}^{\tau}(\mathsf{R},\tilde{m}) = 1\Big) - \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m',f,g}}\Big(\mathrm{D}^{\tau}(\mathsf{R},\tilde{m}) = 1\Big)\right| > \frac{\delta}{2}\right] > \frac{(\delta\delta')^3}{256} - \mathsf{negl},$$

where the outer probability is over $\mathsf{L} \leftarrow \mathcal{L}$ and $(f, g) \leftarrow \mathcal{D}_{\mathsf{split}}$, drawn independently. This breaks the computational conditional augmented non-malleability of $(\mathsf{Enc}, \mathsf{Dec})$, thus completing our proof of Theorem 1.

6 Constructing Codes with Simulatable Right State (Proof of Lemma 1)

Lemma 1 (Restated). $(\mathsf{Enc}, \mathsf{Dec})$ *is computationally conditionally augmented non-malleable against*

$$\mathcal{F}^{\mathsf{poly}}_{\mathsf{split}} = \big\{(f, g) \in \mathcal{F}_{\mathsf{split}} : f \text{ and } g \text{ polytime}\big\}.$$

Fix $(f, g) \in \mathcal{F}^{\mathsf{poly}}_{\mathsf{split}}$ and non-negligible $\delta = \delta(\lambda) > 0$ (every δ in this section refers to this choice). We must describe simulators $\big\{\mathsf{S}^{\mathsf{L}}_{f,g}\big\}_{\mathsf{L}}$ such that for all PPT distinguishers D, $\Pr_{\mathsf{L}}\big[\mathsf{E}_{f,g}\big] < 2\delta + \lambda^{-2}\delta^4 + \mathsf{negl}$, where $\mathsf{E}_{f,g}$ is the event

$$\exists\, m \in \mathcal{M} \text{ st } \left|\Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big) - \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{S}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big)\right| > \delta.$$

We address the three points of the high level proof in Section 3.2 in reverse order. Recall, we already proved Claim 4. Just as in Section 3.2 we use the shorthand $\mathsf{p}_{\mathsf{L},m}$ for $\Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\big(\tilde{m} \neq \perp_{\mathsf{com}}\big)$.

Claim 4 (Restated). *For any $m, m' \in \mathcal{M}$, $(f,g) \in \mathcal{F}_{\mathsf{split}}^{\mathsf{poly}}$ and non-negligible $\xi = \xi(\lambda) > 0$ we have*

$$\Pr_{\mathsf{L}}\Big[\big|\mathsf{p}_{\mathsf{L},m} - \mathsf{p}_{\mathsf{L},m'}\big| > \xi\Big] = \mathsf{negl}.$$

Now, fix any $m^* \in \mathcal{M}$ and define $\mathcal{L}_{\mathsf{VALID}} = \big\{\mathsf{L} \in \mathcal{L} : \mathsf{p}_{\mathsf{L},m^*} \geq \lambda^{-2}\delta^3\big\}$, and $\mathcal{L}_{\mathsf{BOT}} = \mathcal{L} \setminus \mathcal{L}_{\mathsf{VALID}}$. We treat the cases $\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}$ and $\mathsf{L} \in \mathcal{L}_{\mathsf{BOT}}$ separately, beginning with $\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}$. We use (f,g) to construct (f_0,g_0) which tampers $(\mathsf{Enc}_0,\mathsf{Dec}_0)$. We will then use $\mathsf{S}^{\mathsf{L}_0}_{f_0,g_0}$ whose existence is guaranteed by the conditional non-malleability of $(\mathsf{Enc}_0,\mathsf{Dec}_0)$, to construct $\mathsf{S}^{\mathsf{L}}_{f,g}$. Whenever $\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}$, $\mathsf{S}^{\mathsf{L}}_{f,g}$ will simulate $\big\{\mathsf{V}^{\mathsf{L}}_{m,f,g}\big\}_m$. The construction of (f_0,g_0) is as follows.

- **Random Choices:** Draw $k \leftarrow \mathsf{G}(1^\lambda)$, $\sigma \leftarrow \$$, $c^\$ \leftarrow \mathsf{E}_k(0)$, set $z = \mathrm{Com}(k,\sigma)$, and $(\cdot,\tilde{z}^\$) = g(c^\$, z)$.
- $f_0(\mathsf{L}_0)$**:** Compute $\big(\tilde{\mathsf{L}}_0,(\tilde{k},\tilde{\sigma})\big) = f\big(\mathsf{L}_0,(k,\sigma)\big)$. Output $\tilde{\mathsf{L}}_0$.
- $g_0(\mathsf{R}_0)$**:** Draw $c \leftarrow \mathsf{E}_k(\mathsf{R}_0)$ and set $(\tilde{c},\tilde{z}) = g(c,z)$. If $\tilde{z} \neq \tilde{z}^\$$, output $\perp_{\mathsf{com}}$. Otherwise, use superpolynomial time to break open $\tilde{z}$ and recover the pair $(\tilde{k}',\tilde{\sigma}')$. Output $\tilde{\mathsf{R}}_0 = \mathsf{D}_{\tilde{k}'}(\tilde{c})$.

Define $\mathsf{S}^{\mathsf{L}}_{f,g}$ as follows: parse $\mathsf{L} = \big(\mathsf{L}_0,(k,\sigma)\big)$ and draw $c^\$ \leftarrow \mathsf{E}_k(0)$ such that $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})$, where $(\cdot,\tilde{z}^\$) = g\big(c^\$,\mathrm{Com}(k,\sigma)\big)$ and $\big(\tilde{\mathsf{L}}_0,(\tilde{k},\tilde{\sigma})\big) = f(\mathsf{L})$ (we prove in Claim 10 below that such $c^\$$ exist with high probability when $\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}$); this defines $(f_0,g_0) \in \mathcal{F}_{\mathsf{split}}$. Draw $(\mathsf{R}_0,\tilde{m}) \leftarrow \mathsf{S}^{\mathsf{L}_0}_{f_0,g_0}$, $c \leftarrow \mathsf{E}_k(\mathsf{R}_0)$ and output $(\mathsf{R},\tilde{m})$ where $\mathsf{R} = \big(c,\mathrm{Com}(k,\sigma)\big)$.

Claim 10. *For any PPT distinguisher* D, $\Pr_{\mathsf{L}}\big[\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}} \;\&\; \mathsf{E}_{f,g}\big] < \delta + \mathsf{negl}$.

Proof. Assume $\mathcal{L}_{\mathsf{VALID}}$ comprises at least a $\delta-$fraction of $\mathcal{L}$; if not we are done. Note that as $c^\$$ is such that $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})$, tampering with (f_0,g_0) is the same as tampering with (f,g). Indeed, $(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}$ has $\tilde{m} = \perp_{\mathsf{com}}$ if and only if $\mathrm{Decom}(\tilde{z}) \neq (\tilde{k},\tilde{\sigma})$, or equivalently, since Com is perfectly binding, if $\tilde{z} \neq \tilde{z}^\$$. Moreover, if $\tilde{m} \neq \perp_{\mathsf{com}}$ then $g(\mathsf{R}) = \big(\mathsf{E}_{\tilde{k}}(g(\mathsf{R}_0)),\tilde{z}\big)$. It follows that $\mathsf{V}^{\mathsf{L}}_{m,f,g}$ is identical to the distribution: draw $(\mathsf{R}_0,\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}_0}_{m,f_0,g_0}$, set $\mathsf{R} = \big(\mathsf{E}_k(\mathsf{R}_0),\mathrm{Com}(k,\sigma)\big)$ and output $(\mathsf{R},\tilde{m})$. So whenever there exists a $c^\$$ such that $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})$ we have,

$$\Pr_{\mathsf{L}}\big[\mathsf{E}_{f,g}\big] \leq \Pr_{\mathsf{L}}\Big[\exists\, m \text{ st } \Delta\big(\mathsf{V}^{\mathsf{L}_0}_{m,f_0,g_0},\mathsf{S}^{\mathsf{L}_0}_{m,f_0,g_0}\big) > \delta\Big] < \mathsf{negl},$$

as $(\mathsf{Enc}_0,\mathsf{Dec}_0)$ is $\varepsilon-$conditionally augmented non-malleable for negligible ε. Finally, we show that for almost all $\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}$, there exists $c^\$$ st $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})$. In fact, $\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})$ holds for a non-negligible fraction of $c^\$$: $\Pr_{c^\$}\big[\mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})\big|\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}\big] \geq \lambda^{-2}\delta^4 - \mathsf{negl}$. This follows immediately from the definition of $\mathcal{L}_{\mathsf{VALID}}$ and semantic security. Indeed, conditioned on $\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}$, we have that $\Pr_{\mathsf{R}\leftarrow\mathsf{Enc}(m^*|\mathsf{L})}\big(\mathrm{Decom}(\tilde{z}) = (\tilde{k},\tilde{\sigma})\big) \geq \lambda^{-2}\delta^3$, and the only difference between $\mathsf{R} = (c,z) \leftarrow \mathsf{Enc}(m^*|\mathsf{L})$ and $(c^\$,z)$ is their encrypted values. This intuition can be formalized easily using the hiding machine. We complete the proof of Claim 10 by collecting our observations:

$$\begin{aligned}
\Pr_{\mathsf{L}}\big[\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}} \;\&\; \mathsf{E}_{f,g}\big] \;&\leq\; \Pr_{\mathsf{L}}\big[\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}}\big||\mathcal{L}_{\mathsf{VALID}}| < \delta|\mathcal{L}|\big] + \Pr_{\mathsf{L}}\big[\mathsf{E}_{f,g}\big|\exists\, c^\$: \mathrm{Decom}(\tilde{z}^\$) = (\tilde{k},\tilde{\sigma})\big] \\
&+\; \Pr_{\mathsf{L}}\big[\mathrm{Decom}(\tilde{z}^\$) \neq (\tilde{k},\tilde{\sigma})\;\forall\, c^\$\big|\mathsf{L} \in \mathcal{L}_{\mathsf{VALID}} \;\&\; |\mathcal{L}_{\mathsf{VALID}}| \geq \delta|\mathcal{L}|\big] \\
&<\; \delta + \mathsf{negl}.
\end{aligned}$$

□

When $\mathsf{L} \in \mathcal{L}_{\mathsf{BOT}}$, (f,g) is almost always tampering to $\perp_{\mathsf{com}}$. In this case, computational conditional augmented non-malleability follows from right state simulatability. We define $\big\{\mathsf{S}^{\mathsf{L}}_{f,g}\big\}_{\mathsf{L}}$ simply: $\mathsf{S}^{\mathsf{L}}_{f,g}$ draws $\mathsf{R} \leftarrow \mathcal{D}_{\mathsf{hid}}$ (the right state simulator) and outputs $(\mathsf{R},\perp_{\mathsf{com}})$.

Claim 11. *For any PPT distinguisher* D, $\Pr_{\mathsf{L}}\big[\mathsf{L} \in \mathcal{L}_{\mathsf{BOT}} \;\&\; \mathsf{E}_{f,g}\big] < \delta$.

Proof. We use the hiding machine. Set BAD to be the set of $\mathsf{L} \in \mathcal{L}_{\mathsf{BOT}}$ for which $\mathsf{E}_{f,g}$ occurs. Unless L is in the negligible fraction of $\mathcal{L}$ for which Claim 4 does not hold, $\mathsf{p}_{\mathsf{L},m} < 2\lambda^{-2}\delta^3$ (using Claim 4 with $\xi = \lambda^{-2}\delta^3$). For such L, we have

$$\begin{aligned}\Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1 \big| \tilde{m} = \perp_{\mathsf{com}}\big) &\geq \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big) - 2\lambda^{-2}\delta^3 \\ &> \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{S}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big) + \delta - 2\lambda^{-2}\delta^3 \\ &> \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{S}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big) + \frac{\delta}{2},\end{aligned}$$

for some $m \in \mathcal{M}$. Assume for contradiction that $\Pr_{\mathsf{L}}\big[\mathsf{L} \in \mathsf{BAD}\big] \geq \delta$. Fix $N = 5/\delta$, $N' = \Omega(\lambda\delta^{-2})$ and consider the PPT adversary $\mathcal{A}$ who plays with challenger $\mathcal{C}$ as follows.

- $\mathcal{C}$ chooses $\mathsf{L} \leftarrow \mathcal{L}$, $i^* \leftarrow \{1,\ldots,N\}$, sends $\big(\{\mathsf{R}_1\},\ldots,\{\mathsf{R}_N\}\big)$ to $\mathcal{A}$ where $\mathsf{R}_i \leftarrow \mathcal{D}_{\mathsf{hid}}$ for all $i \neq i^*$, $\mathsf{R}_i \in \{\mathsf{R}_i\}$; and $\mathsf{R}_{i^*} \leftarrow \mathsf{Enc}(m|\mathsf{L})$ for all $\mathsf{R}_{i^*} \in \{\mathsf{R}_{i^*}\}$.
- $\mathcal{A}$ receives $\big(\{\mathsf{R}_1\},\ldots,\{\mathsf{R}_N\}\big)$ and for $i = 1,\ldots,N$, sets $p_i = \Pr_{\mathsf{R}_i\in\{\mathsf{R}_i\}}\big(\mathrm{D}(\mathsf{R},\perp_{\mathsf{com}}) = 1\big)$ and outputs i^* such that p_{i^*} is maximal.

If $\mathsf{L} \in \mathcal{L}$ chosen by $\mathcal{C}$ is in BAD and such that Claim 4 holds then the expected number of (i,R_i) such that $i \in \{1,\ldots,N\}$, $\mathsf{R}_i \in \{\mathsf{R}_i\}$, and $\tilde{m}_i \neq \perp_{\mathsf{com}}$ is at most $2\lambda^{-2}\delta^3 NN' < 1/2$, where $\tilde{m}_i = \mathsf{Dec}(\tilde{\mathsf{L}},\tilde{\mathsf{R}}_i)$. With probability at least $1/2$ there exist no such (i,R_i). In this case we have

$$p_{i^*} \geq \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big|\tilde{m} = \perp_{\mathsf{com}}\big) - \frac{\delta}{6} > \Pr_{(\mathsf{R},\tilde{m})\leftarrow \mathsf{S}^{\mathsf{L}}_{f,g}}\big(\mathrm{D}(\mathsf{R},\tilde{m}) = 1\big) + \frac{\delta}{2} - \frac{\delta}{6} \geq p_i + \frac{\delta}{6},$$

for all $i \neq i^*$. So we see that

$$\Pr\big(\mathcal{A}\text{ wins}\big) \geq \Pr\big(\mathsf{L} \in \mathsf{BAD}\big)\Pr\big(\tilde{m}_i = \perp_{\mathsf{com}}\ \forall\ (i,\mathsf{R}_i)\big|\mathsf{L} \in \mathsf{BAD}\big) - \mathsf{negl} \geq \frac{\delta}{2} - \mathsf{negl} > \frac{2}{N},$$

and so $\mathcal{A}$ breaks the right state simulatability of $(\mathsf{Enc},\mathsf{Dec})$. □

7 The Extended Protocol

In this section we extend the protocol of Section 4 so it remains non-malleable against a non-synchronizing adversary. The only non-synchronizing scheduling available to the adversary which is not trivially dealt with is the sequential scheduling where he lets the left interaction complete before beginning the right. We protect against such an adversary by relying on extraction. We provide two separate constructions with different extractability guarantees. The first is more lightweight, using the underlying commitment as a blackbox, and admits an extractor which extracts $\mathcal{C}$'s commitment with non-negligible probability. While sufficient for non-malleability, often in applications it is desirable to have an extractor which can recover $\mathcal{C}$'s commitment with overwhelming probability. We give a second construction which admits such an extractor, but makes non-blackbox use of the commitment scheme. Our second construction is public coin, while our first is not.

7.1 The Blackbox Construction

Our first construction is shown in Figure 4.

Theorem 2. $\langle \mathcal{C},\mathcal{R}\rangle_{\mathsf{nm}}$ *is a perfectly binding, non-malleable commitment scheme.*

Setup: Let Com be a non-interactive, perfectly binding commitment scheme. Let $(\mathsf{Enc}, \mathsf{Dec})$ be a computational, conditional, augmented non-malleable code. Fix a large prime q, let $\mathsf{id} \in \{0,1\}^\lambda$ be $\mathcal{C}$'s identity.

Commiter's Private Input: $v \in \mathcal{M}_{\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}}$ to be committed to.

Commit Phase:

1. $\mathcal{C} \to \mathcal{R}$**:**
   - Set $m = v \circ \mathsf{id}$ and draw $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m)$, where $\mathsf{L} \in \mathcal{L} \subset \mathbb{Z}_q$. Choose random $r \leftarrow \mathbb{Z}_q$, $\omega \leftarrow \$$ send $\mathrm{Com}(\mathsf{L} \circ r; \omega)$ to $\mathcal{R}$.
   - Set $\mathsf{X} = (\mathsf{L} \circ r \circ \omega) \in \mathbb{Z}_{q'}$; choose a degree $\lambda/2$ polynomial $p(x) \in \mathbb{Z}_{q'}[x]$ such that $p(0) = \mathsf{X}$ and for $i = 1, \ldots, \lambda$ set $\mathsf{X}_i = p(i)$. Choose $\mathsf{X}_i^0, \mathsf{X}_i^1, \mathsf{X}_i^2 \in \mathbb{Z}_{q'}$ randomly such that $\mathsf{X}_i^0 + \mathsf{X}_i^1 + \mathsf{X}_i^2 = \mathsf{X}_i$. Send $\mathsf{Y}_i^b = \mathrm{Com}(\mathsf{X}_i^b)$, to $\mathcal{R}$ for $b \in \{0,1,2\}$.
2. $\mathcal{R} \to \mathcal{C}$**:** Send random challenge $\alpha \in \mathbb{Z}_q^*$, $\mathsf{c} \in \{0,1,2\}^\lambda$, and commitment $\mathrm{Com}(S)$ for a random subset $S \subset \{1, \ldots, \lambda\}$ of size $\lambda/10$.
3. $\mathcal{C} \to \mathcal{R}$**:** Send $a = r\alpha + \mathsf{L}$, R and $\mathrm{Decom}(\mathsf{Y}_i^{\mathsf{c}_i})$ for $i = 1, \ldots, \lambda$.

Decommit Phase:

1. $\mathcal{C} \to \mathcal{R}$**:** Open the commitments sent in step 1. Let $\mathsf{L}', r' \in \mathbb{Z}_q$ and $\mathsf{X}_i^b \in \{0,1\}^{\mathsf{poly}(\lambda)}$ be the decommitted values.

Receiver's Output: If any of the decommitments sent in step 3 of the commit phase fail, output $\perp$. Otherwise, set $\mathsf{X}_i' = \mathsf{X}_i^0 + \mathsf{X}_i^1 + \mathsf{X}_i^2$, where X_i^b are the values decommitted to in the decommitment phase. If any X_i^b is ill formed, set $\mathsf{X}_i' = 0$. Use error correction to recover a polynomial $f(x) \in \mathbb{Z}_{q'}[x]$ of degree at most $\lambda/2$ such that $f(i) = \mathsf{X}_i'$ for at least $3\lambda/4$ values of i. If no such $f(x)$ exists, output $\perp$. Now check that $f(i) = \mathsf{X}_i'$ for all $i \in S$, if not output $\perp$. Otherwise, let $\mathsf{X} = f(0)$. If X is not a valid decommitment to $\mathrm{Com}(\mathsf{L} \circ r; \omega)$, sent in step one output $\perp_{\mathsf{fail}}$. If L and r do not satisfy $r\alpha + \mathsf{L} = a$ then output $\perp_{\mathsf{inc}}$. Otherwise, compute $m = \mathsf{Dec}(\mathsf{L}, \mathsf{R})$ and parse $m = v \circ \mathsf{id}'$. Output v if $\mathsf{id}' = \mathsf{id}$, $\perp_{\mathsf{id}}$ if not.

Figure 3: Non-malleable commitment scheme $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$.

Proof. Perfect binding follows from the perfect binding of Com. Hiding follows from a hybrid argument: first change the $(\mathsf{X}_i^0, \mathsf{X}_i^1, \mathsf{X}_i^2)$ one by one from random subject to $\mathsf{X}_i^0 + \mathsf{X}_i^1 + \mathsf{X}_i^2 = \mathsf{X}_i$ to random; then use the hiding of Com and the fact that any split-state non-malleable code is also a 2-out-of-2 secret sharing scheme.

To prove non-malleability against a sequential adversary we construct an extractor E which extracts M's commitment with non-negligible probability over $\mathcal{R}$'s randomness (so success of extraction is independent of M's commitment). Therefore, if M mauls the commitment with non-negligible probability, then with related probability he is mauling *and* E succeeds in extracting M's value. This breaks hiding. Now, more formally. Suppose M mauls $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ with probability $\delta > 0$. E works as follows. E receives a left transcript and first message of a right session as input. E chooses $\tilde{\alpha} \in \mathbb{Z}_q^*$ at random and does the following polynomially many times (δ^{-3} suffices): draw $\tilde{\mathsf{c}}' \leftarrow \{0,1,2\}^\lambda$ and $\tilde{S}' \subset \{1, \ldots, \lambda\}$ at random and send $\big(\tilde{\alpha}, \tilde{\mathsf{c}}', \mathrm{Com}(\tilde{S})\big)$, and receive answers from M. Let $\mathsf{REC} \subset \{1, \ldots, \lambda\}$ be the set of i for which M correctly decommits to $\tilde{\mathsf{X}}_i^0$, $\tilde{\mathsf{X}}_i^1$, $\tilde{\mathsf{X}}_i^2$ during the polynomially many rewinds. There exists a constant C such that $|\mathsf{REC}| \geq \lambda - C \log \lambda$ with overwhelming probability. E sets $\tilde{\mathsf{X}}_i = \tilde{\mathsf{X}}_i^0 + \tilde{\mathsf{X}}_i^1 + \tilde{\mathsf{X}}_i^2$ for $i \in \mathsf{REC}$ and error corrects recovering $\tilde{\mathsf{X}}$. If error correction fails or if $\tilde{\mathsf{X}}$ is an invalid decommitment to $\tilde{\mathsf{L}} \circ \tilde{r}$ then E outputs FAIL. Otherwise, E one more time chooses $(\tilde{c}, \tilde{S})$,

sends $\big(\tilde{\alpha}, \tilde{c}, \mathrm{Com}(\tilde{S})\big)$ and receives M's third message. If M's third message is malformed, or if $\tilde{\alpha}$ is answered incorrectly, or if $\tilde{S} \not\subset \mathsf{REC}$ E outputs FAIL. Otherwise, E outputs $\tilde{m}$ (using the $\tilde{\mathsf{L}}$ it extracted and $\tilde{\mathsf{R}}$ send in M's final third message).

Note that whenever E outputs $\tilde{m}$, M has committed to $\tilde{m}$ on the right. Therefore, it suffices to show that with non-negligible probability, E extracts $\tilde{m}$ from a transcript which M has mauled. In order for M to prevent this, he must choose to maul only transcripts for which $\tilde{S} \not\subset \mathsf{REC}$. However, such an M breaks the hiding of Com used to commit to $\tilde{S}$, as follows. Suppose M is such that

Finally, the proof of non-malleability against a synchronizing adversary roughly follows from the same argument as for the basic protocol. We construct tampering functions (f, g) which maul the code if M mauls the commitment. The main difference is that partial transcript (f, g) share now includes the commitments to the X_i^b. However, f rewinds M only once, and each X_i is split into three shares, so this causes no problem. We provide a sketch. If M breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ then there exist $m, m' \in \mathcal{M}$ and non-negligible $\delta > 0$ st $\Pr_\tau\big[\tau \in \mathsf{MAUL}_{m,m'}\big] \geq \delta/2$, where $\tau \in \mathsf{MAUL}_{m,m'}$ if

$$\left| \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{M}^\tau_m} \Big(D^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{M}^\tau_{m'}} \Big(D^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) \right| \geq \frac{\delta}{2}.$$

The partial transcript τ includes the first two rounds of the right execution and the entire left execution except for R. As before, τ is completed to a full transcript once R is specified. Note that if M's commitment $\tilde{v}$ is to $\perp_{\mathsf{fail}}$, then M's first message is bad, either because $\tilde{\mathsf{X}}_i = \tilde{\mathsf{X}}_i^0 + \tilde{\mathsf{X}}_i^1 + \tilde{\mathsf{X}}_i^2$ are not close enough to the valuations of a polynomial, or because this polynomial's constant term is not a valid decommitment of $\mathrm{Com}(\tilde{\mathsf{L}} \circ \tilde{r}; \omega)$. In any case, M cannot be mauling if $\tilde{v} = \perp_{\mathsf{fail}}$ as $\mathcal{C}$'s commitment on the left is not even defined after the first message. Also, the proof of Lemma 2 goes through unchanged for this new protocol and so it follows that if M is mauling with non-negligible probability then he is doing so while also sending the correct value for $\tilde{a}$ with non-negligible probability.

This allows us to build a distribution $\mathcal{D}_{\mathrm{M}}$ on tampering functions which we will use to break the security of $(\mathsf{Enc}, \mathsf{Dec})$. As before $(f, g) \leftarrow \mathcal{D}_{\mathrm{M}}$ share a random partial transcript τ and a random $\mathsf{R}_\$$, let $\tilde{a}_\$$ be M's response on the right in the transcript $\mathbb{T}(\tau, \mathsf{R}_\$)$. $f(\mathsf{L})$ extracts $\tilde{\mathsf{L}}$ by rewinding M and asking a new challenge $\tilde{\beta}$ on the right, using L and τ to answer on the left. We provide f with the decommitments $\mathrm{Decom}(\mathsf{Y}_i^b)$ so he can answer this part of M's query on the left honestly. $g(\mathsf{R})$ completes τ to $\mathbb{T}(\tau, \mathsf{R})$ and checks whether M's answer $\tilde{a}$ is equal to $\tilde{a}_\$$ or not. If so $g(\mathsf{R}) = \tilde{\mathsf{R}}$, where $\tilde{\mathsf{R}}$ is from M's final message of $\mathbb{T}(\tau, \mathsf{R})$. If not $g(\mathsf{R}) = \perp_{\mathsf{inc}}$. The same proof of Lemma 3 shows that if M breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ then there exist $m, m' \in \mathcal{M}$ and non-negligible $\delta, \delta' > 0$ such that

$$\Pr_{\mathsf{L},(f,g)} \left[\left| \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}} \Big(\mathrm{D}^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m',f,g}} \Big(\mathrm{D}^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) \right| > \frac{\delta}{2} \right] > \delta', \tag{3}$$

where the outer probability is over $\mathsf{L} \leftarrow \mathcal{L}$ and $(f, g) \leftarrow \mathcal{D}_{\mathrm{M}}$ using randomness which depends on L. In order to complete the reduction to the security of $(\mathsf{Enc}, \mathsf{Dec})$ we need to exhibbit a distribution $\mathcal{D}_{\mathsf{split}}$ on split state tampering function pairs such that Equation 4 holds when $(f, g) \leftarrow \mathcal{D}_{\mathsf{split}}$. In the proof of Theorem 1, we used a simple hybrid argument to argue that $\mathcal{D}_{\mathrm{M}} \approx_{\mathsf{c}} \mathcal{D}_{\mathsf{split}}$ and so (5) followed straight from (4). We argue similarly, except our first hybrid is to change the decommitment information $\big\{\mathrm{Decom}(\mathsf{Y}_i^b)\big\}$ given to f. Instead of the Y_i^b being commitments to X_i^b such that $\mathsf{X}_i^0 + \mathsf{X}_i^1 + \mathsf{X}_i^2 = \mathsf{X}_i$ where $\mathsf{X}_i = p(i)$ for a polynomial $p(i)$, we change Y_i^b to be commitments to uniformly random X_i^b. Just like in the hybrid argument for hiding, these changes can be made one by one and the output distribution of $\tilde{m}$ cannot change since f uses just two of the three values X_i^b for all i. The remainder of the hybrid argument goes through exactly as for the basic protocol. □

7.2 The Extractable Construction

Let $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$ be the commitment of Section 4 and let $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{ext}}$ be a (malleable) three round extractable commitment scheme. Our commitment scheme in this section, $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ commits to v as follows: it uses $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}$

to commit to v while, in parallel, using $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{ext}}$ to commit to the decommitment information of the first commitment. We prove that this composition enjoys the best of both of its building blocks: it is extractable (and so non-malleable against a sequential adversary) while still being non-malleable against a synchronizing adversary. One technical point is that in the proof of synchronizing non-malleability, we need to rewind the protocol one time, therefore to make our proof go through, we need extraction from $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{ext}}$ to require two rewinds. The protocol is shown in Figure 4.

Setup: Let Com be a non-interactive, perfectly binding commitment scheme. Let $(\mathsf{Enc}, \mathsf{Dec})$ be a computational, conditional, augmented non-malleable code. Fix a large prime q, let $\mathsf{id} \in \{0,1\}^\lambda$ be $\mathcal{C}$'s identity.

Commiter's Private Input: $v \in \mathcal{M}_{\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{synch}}}$ to be committed to.

Commit Phase:

1. $\mathcal{C} \to \mathcal{R}$**:**
   - Set $m = v \circ \mathsf{id}$ and draw $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m)$, where $\mathsf{L} \in \mathcal{L} \subset \mathbb{Z}_q$. Choose random $r \leftarrow \mathbb{Z}_q$, $\omega \leftarrow \$$ send $\mathrm{Com}(\mathsf{L} \circ r; \omega)$ to $\mathcal{R}$.
   - Set $\mathsf{X} = (\mathsf{L} \circ r \circ \omega) \in \mathbb{Z}_{q'}$; choose a degree $\lambda/2$ polynomial $p(x) \in \mathbb{Z}_{q'}[x]$ such that $p(0) = \mathsf{X}$ and for $i = 1, \ldots, \lambda$ set $\mathsf{X}_i = p(i)$. Choose $\mathsf{X}_i^0, \mathsf{X}_i^1, \mathsf{X}_i^2 \in \mathbb{Z}_{q'}$ randomly such that $\mathsf{X}_i^0 + \mathsf{X}_i^1 + \mathsf{X}_i^2 = \mathsf{X}_i$. Send $\mathsf{Y}_i^b = \mathrm{Com}(\mathsf{X}_i^b)$, to $\mathcal{R}$ for $b \in \{0,1,2\}$.
2. $\mathcal{R} \to \mathcal{C}$**:** Send random challenge $\alpha \in \mathbb{Z}_q^*$, $\mathsf{c} \in \{0,1,2\}^\lambda$, and commitment $\mathrm{Com}(S)$ for a random subset $S \subset \{1, \ldots, \lambda\}$ of size $\lambda/10$.
3. $\mathcal{C} \to \mathcal{R}$**:** Send $a = r\alpha + \mathsf{L}$, R and $\mathrm{Decom}(\mathsf{Y}_i^{\mathsf{c}_i})$ for $i = 1, \ldots, \lambda$.

Decommit Phase:

1. $\mathcal{C} \to \mathcal{R}$**:** Open the commitments sent in step 1. Let $\mathsf{L}', r' \in \mathbb{Z}_q$ and $\mathsf{X}_i^b \in \{0,1\}^{\mathsf{poly}(\lambda)}$ be the decommitted values.

Receiver's Output: If any of the decommitments sent in step 3 of the commit phase fail, output $\perp$. Otherwise, set $\mathsf{X}_i' = \mathsf{X}_i^0 + \mathsf{X}_i^1 + \mathsf{X}_i^2$, where X_i^b are the values decommitted to in the decommitment phase. If any X_i^b is ill formed, set $\mathsf{X}_i' = 0$. Use error correction to recover a polynomial $f(x) \in \mathbb{Z}_{q'}[x]$ of degree at most $\lambda/2$ such that $f(i) = \mathsf{X}_i'$ for at least $3\lambda/4$ values of i. If no such $f(x)$ exists, output $\perp$. Now check that $f(i) = \mathsf{X}_i'$ for all $i \in S$, if not output $\perp$. Otherwise, let $\mathsf{X} = f(0)$. If X is not a valid decommitment to $\mathrm{Com}(\mathsf{L} \circ r; \omega)$, sent in step one output $\perp_{\mathsf{fail}}$. If L and r do not satisfy $r\alpha + \mathsf{L} = a$ then output $\perp_{\mathsf{inc}}$. Otherwise, compute $m = \mathsf{Dec}(\mathsf{L}, \mathsf{R})$ and parse $m = v \circ \mathsf{id}'$. Output v if $\mathsf{id}' = \mathsf{id}$, $\perp_{\mathsf{id}}$ if not.

Figure 4: Non-malleable commitment scheme $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$.

Claim 12. *$\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ is a perfectly binding extractable commitment scheme.*

Proof Sketch. Perfect binding follows from the perfect binding of Com. Computational hiding follows from the hiding of Com and the fact that any split-state non-malleable code is also a 2-out-of-2 secret sharing scheme. Finally, extractability follows from a standard argument about the extractability of the subprotocol we've added to the basic scheme; see [CDMW08] for a similar argument. □

Theorem 3. *$\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ is non-malleable.*

Proof. It suffices to prove that it is non-malleable against a synchronizing adversary as the only other non-trivial scheduling is the sequential one and non-malleability against a sequential adversary follows from extractability. We follow the same outline and use the same notation as in the proof of Theorem 1. Recall that if M breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ then there exist $m, m' \in \mathcal{M}$ and non-negligible $\delta > 0$ st $\Pr_\tau\left[\tau \in \mathsf{MAUL}_{m,m'}\right] \geq \delta/2$, where $\tau \in \mathsf{MAUL}_{m,m'}$ if

$$\left| \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{M}^\tau_m} \Big(D^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{M}^\tau_{m'}} \Big(D^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) \right| \geq \frac{\delta}{2}.$$

The partial transcript τ includes the first two rounds of the right execution and the entire left execution except for R. As before, τ is completed to a full transcript once R is specified. Note that if M's commitment $\tilde{v}$ is to $\perp_{\mathsf{fail}}$, then M's first message is bad, either because $\tilde{\mathsf{X}}_i = \tilde{\mathsf{X}}^0_i + \tilde{\mathsf{X}}^1_i + \tilde{\mathsf{X}}^2_i$ are not close enough to the valuations of a polynomial, or because this polynomial's constant term is not a valid decommitment of $\mathrm{Com}(\tilde{\mathsf{L}} \circ \tilde{r}; \omega)$. In any case, M cannot be mauling if $\tilde{v} = \perp_{\mathsf{fail}}$ as $\mathcal{C}$'s commitment on the left is not even defined after the first message. Also, the proof of Lemma 2 goes through unchanged for this new protocol and so it follows that if M is mauling with non-negligible probability then he is doing so while also sending the correct value for $\tilde{a}$ with non-negligible probability.

This allows us to build a distribution $\mathcal{D}_{\mathrm{M}}$ on tampering functions which we will use to break the security of $(\mathsf{Enc}, \mathsf{Dec})$. As before $(f, g) \leftarrow \mathcal{D}_{\mathrm{M}}$ share a random partial transcript τ and a random $\mathsf{R}_\$$, let $\tilde{a}_\$$ be M's response on the right in the transcript $\mathbb{T}(\tau, \mathsf{R}_\$)$. $f(\mathsf{L})$ extracts $\tilde{\mathsf{L}}$ by rewinding M and asking a new challenge $\tilde{\beta}$ on the right, using L and τ to answer on the left. We provide f with the decommitments $\mathrm{Decom}(\mathsf{Y}^b_i)$ so he can answer this part of M's query on the left honestly. $g(\mathsf{R})$ completes τ to $\mathbb{T}(\tau, \mathsf{R})$ and checks whether M's answer $\tilde{a}$ is equal to $\tilde{a}_\$$ or not. If so $g(\mathsf{R}) = \tilde{\mathsf{R}}$, where $\tilde{\mathsf{R}}$ is from M's final message of $\mathbb{T}(\tau, \mathsf{R})$. If not $g(\mathsf{R}) = \perp_{\mathsf{inc}}$. The same proof of Lemma 3 shows that if M breaks the non-malleability of $\langle \mathcal{C}, \mathcal{R} \rangle_{\mathsf{nm}}$ then there exist $m, m' \in \mathcal{M}$ and non-negligible $\delta, \delta' > 0$ such that

$$\Pr_{\mathsf{L},(f,g)} \left[\left| \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}} \Big(\mathrm{D}^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m',f,g}} \Big(\mathrm{D}^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) \right| > \frac{\delta}{2} \right] > \delta', \tag{4}$$

where the outer probability is over $\mathsf{L} \leftarrow \mathcal{L}$ and $(f, g) \leftarrow \mathcal{D}_{\mathrm{M}}$ using randomness which depends on L. In order to complete the reduction to the security of $(\mathsf{Enc}, \mathsf{Dec})$ we need to exhibbit a distribution $\mathcal{D}_{\mathsf{split}}$ on split state tampering function pairs such that

$$\Pr_{\mathsf{L},(f,g)} \left[\left| \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m,f,g}} \Big(\mathrm{D}^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) - \Pr_{(\mathsf{R},\tilde{m}) \leftarrow \mathsf{V}^{\mathsf{L}}_{m',f,g}} \Big(\mathrm{D}^\tau(\mathsf{R}, \tilde{m}) = 1 \Big) \right| > \frac{\delta}{2} \right] > \delta', \tag{5}$$

where the outer probability is over $\mathsf{L} \leftarrow \mathcal{L}$ and $(f, g) \leftarrow \mathcal{D}_{\mathsf{split}}$ drawn independently. In the proof of Theorem 1, we used a simple hybrid argument to argue that $\mathcal{D}_{\mathrm{M}} \approx_{\mathsf{c}} \mathcal{D}_{\mathsf{split}}$ and so (5) followed straight from (4). We argue similarly, except our first hybrid is to change the decommitment information $\left\{\mathrm{Decom}(\mathsf{Y}^b_i)\right\}$ given to f. Instead of the Y^b_i being commitments to X^b_i such that $\mathsf{X}^0_i + \mathsf{X}^1_i + \mathsf{X}^2_i = \mathsf{X}_i$ where $\mathsf{X}_i = p(i)$ for a polynomial $p(i)$, we change Y^b_i to be commitments to uniformly random X^b_i. Note that f rewinds M once and so the output of the tampering distribution $\mathsf{T}^{\mathsf{L}}_{m,f,g}$ is identical after this change is made. Indeed, f uses just two of the three values X^b_i for all i which are not contained in one of the sets S, M queries. Just two out of $\mathsf{X}^0_i, \mathsf{X}^1_i, \mathsf{X}^2_i$ is random either way. Moreover, as $|S| = \lambda/10$, f uses at most $\lambda/5$ valuations of $p(i)$ in the two rewinds and $\deg(p) = \lambda/2 > \lambda/5$, so these values are uniformly random in both cases. The remainder of the hybrid argument goes through exactly as for the basic protocol. □

8 Simple Constant-Round Non-malleable Commitments

In this section, we describe a very simple non-malleable commitment scheme based on standard split-state non-malleable codes. The scheme has constant rounds and an almost elementary proof. Let us quickly recall the

notation for Naor's statistically binding commitment scheme[Nao91] based on a pseudorandom generator $\mathsf{prg} : \{0,1\}^{\lambda} \to \{0,1\}^{3\lambda}$ (which can be constructed from any one-way function [HILL99]) and Feige-Shamir zero-knowledge proof-of-knowledge [FS89].

Naor's commitment consists of two rounds: first the receiver sends a random string τ of length 3λ; to commit to a bit b, the sender sends $c = \mathsf{com}_{\tau}(b)$ as follows: select a uniform seed $r \in \{0,1\}^{\lambda}$ and set $c = \mathsf{prg}(r)$ if $b = 0$, and $c = \tau \oplus \mathsf{prg}(r)$ if $b = 1$. To decommit, or "open" c, the sender simply provides r. The receiver accepts 0 as the decommitted value if $c = \mathsf{prg}(r)$, 1 if $c = \tau \oplus \mathsf{prg}(r)$, and rejects otherwise. If τ is not of the form $\mathsf{prg}(r) \oplus \mathsf{prg}(r')$, which happens with 2^{-n} probability, the scheme is perfectly binding. When we want to be explicit about the randomness used for commitment we write $c = \mathsf{com}_{\tau}(b; r)$.

The protocol has the following **equivocality** property: if τ is of the special form $\tau = \mathsf{prg}(r) \oplus \mathsf{prg}(r')$ then by setting $c = \mathsf{prg}(r)$, the committer can open c to both 0 and 1 by sending either r or r' respectively. A *string* $s = (s_1, \ldots, s_n)$ can be committed bit by bit: the receiver selects n strings $\tau_1, \ldots, \tau_n$, and the sender sends $\mathsf{com}_{\tau_i}(s_i; r_i)$ for every $i \in [n]$. We abuse the notation and denote this commitment to s by $\mathsf{com}_{\tau}(s; r) := (\mathsf{com}_{\tau_1}(s_1; r_1), \ldots, \mathsf{com}_{\tau_n}(s_n; r_n))$ where $\tau = (\tau_1, \ldots, \tau_n)$ and $r = (r_1, \ldots, r_n)$. Equivocality for s is achieved through equivocality for each s_i when every τ_i is of special form. When equivocality is not desired, the same string, e.g., τ_1, can be used for all bits. It will be clear from the context if we want to use a single string for all bits, or a different one for each bit.

Feige-Shamir zero-knowledge proof-of-knowledge [FS89] will be denoted by Π_{FS}. For concreteness, we use the four round variant of [GRRV14], and denote its messages by $(\mathsf{zk}_1, \mathsf{zk}_2, \mathsf{zk}_3, \mathsf{zk}_4)$. This protocol has the property the statement to be proven can be decided in the last round. It is based on the three round protocols of [Blu86, LS90] and inherits their "special soundness" and "proof-of-knowledge" properties; furthermore, the second round of the protocol consists of λn^2 bit-commitments where n is the number of nodes in the graph corresponding to the statement being proven. For concreteness, we denote the "knowledge" extractor of Π_{FS} by Ext and the zero-knowledge simulator by ZKSim. Without loss of generality, we assume that Ext^{P^*}, w.r.t. any cheating prover P^*, works by rewinding P^* only in last two messages: it rewinds P^* until it obtains two accepting transcripts from which a suitable witness can be obtained; the expected running time of Ext^{P^*} is $\mathsf{poly}(\lambda)/p$ where p is the convincing probability of P^* w.r.t. an appropriate zk_3.

Our commitment protocol $\langle \mathcal{C}, \mathcal{R} \rangle$. We now describe our protocol for committing messages of length n and supporting tags of length t where n, t can be arbitrary polynomials in the security parameter λ. Let $(\mathsf{Enc}, \mathsf{Dec})$ be an efficient split-state non-malleable code for messages of length $n + t$ against the class $\mathcal{F}_{\mathsf{split}}$. For simplicity, we assume that the states are of length ℓ, and tampering functions map from $\{0,1\}^{\ell}$ to $\{0,1\}^{\ell}$. Let com and Π_{FS} be Naor's commitment and Feige-Shamir protocols respectively as defined above.

The input to the commitment algorithm, $\mathcal{C}$, is a message $m \in \{0,1\}^n$ and an identity $\mathsf{id} \in \{0,1\}^t$. The protocol proceeds in following stages:

Stage 1. $\mathcal{C}$ and $\mathcal{R}$ perform a coin-tossing protocol.

(a) $\mathcal{R}$ sends a random $\tau \leftarrow \{0,1\}^{3\lambda}$ (the first message of Naor's commitment)
(b) $\mathcal{C}$ commits to a random string ρ' i.e., $c = \mathsf{com}_{\tau}(\rho'; r)$
(c) $\mathcal{R}$ sends a random string ρ''
(d) $\mathcal{C}$ sends ρ'
(e) $\mathcal{C}$ proves that: "$\exists r : c = \mathsf{com}_{\tau}(\rho'; r)$" using protocol Π_{FS}.
    It uses com_{τ} as the commitment scheme for *every* commitment in Π_{FS}.
    The transcript of this execution of Π_{FS} is denoted by $(\mathsf{zk}_1, \mathsf{zk}_2, \mathsf{zk}_3, \mathsf{zk}_4)$.
(f) Define $\rho = \rho' \oplus \rho''$, and parse $\rho := \rho_1 \| \rho_{\mathsf{FS}} \| \rho_2$

Stage 2. $\mathcal{C}$ encodes the message $m \| \mathsf{id}$, and commits to the first state *using* ρ_1, i.e.,

(a) $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m \| \mathsf{id}; v)$

(b) Send $c_1 \leftarrow \mathsf{com}_{\rho_1}(\mathsf{L}; v_1)$

Stage 3. $\mathcal{C}$ and $\mathcal{R}$ start a fresh execution of Π_{FS} where the statement x to be proven is decided in the last round. Let G_x denote the Hamiltonian graph corresponding to x, and q be the number of nodes in G_x. Com and $\mathcal{R}$ complete the first three rounds of Π_{FS}, denoted $(\mathsf{zk}'_1, \mathsf{zk}'_2, \mathsf{zk}'_3)$ where every commitment in α'_2 is performed using a *unique part* of ρ_{FS}.[2] In the final round, $\mathcal{C}$ sends the following:

(a) Commitment to the second state: $c_2 \leftarrow \mathsf{com}_{\rho_2}(\mathsf{R}; v_2)$, and

(b) zk'_4 which is the last message of Π_{FS} proving the following statement x:

"$\exists(\mathsf{L}, \mathsf{R}, v_1, v_2, m) : c_1 = \mathsf{com}_{\rho_1}(\mathsf{L}; v_1) \wedge c_2 = \mathsf{com}_{\rho_2}(\mathsf{R}; v_2) \wedge \mathsf{Dec}(\mathsf{L}, \mathsf{R}) = m \| \mathsf{id}$."

Stage 4. Com "opens" c_2 to R by sending v_2.

To open the commitment, the committer simply reveals v_1 (and hence L). The message is recovered by decoding L and R (available as part of the commitment transcript).

This concludes the description of our commitment scheme. For concreteness, let us note that the size of ρ_1, ρ_2 is $3\lambda\ell$ each, size of statement x in 3(b) is $6\ell\lambda + t$ and its witness has length $(6\lambda+2)\ell + n$, $q = q(\lambda, n, t, \ell)$ is a fixed polynomial defined by the NP-reduction from x to G_x, and ρ_{FS} has length $3\lambda \cdot \lambda q^2$. Strings ρ, ρ', ρ'' thus have length $3\lambda(2\ell + \lambda q^2)$ each.

Theorem 4. *Protocol $\langle \mathcal{C}, \mathcal{R} \rangle$ is a tag-based statistically-binding commitment scheme that is non-malleable with respect to commitments for messages and tags of arbitrary polynomial length.*

Proof. The scheme is statistically-binding for every efficient committers due to the fact that com_τ is statistically binding and protocol Π_{FS} is sound. It is computationally hiding because com_{ρ_1} and com_{ρ_2} are computationally hiding and the protocol Π_{FS} in stage 4 is zero-knowledge.

We now prove that the scheme is non-malleable w.r.t. commitments. We divide the proof in two parts: in Section 8.1, lemma 4, we prove that the scheme is non-malleable w.r.t. synchronous strategies; in Section 8.2, lemma 5, we prove that it is also non-malleable w.r.t. every M who employs any scheduling except synchronous. It follows that the scheme is non-malleable w.r.t. all adversaries since any successful M must also succeed, with at least half the advantage, by following one of these strategies. □

8.1 Non-malleability against Synchronous Strategies

Proof overview. As discussed earlier, the proof will consider several hybrid experiments and create a situation where the commitments of man-in-the-middle to the two states can be separated into two tampering functions which tamper the states independently. This will be done by eventually obtaining a transcript that is statistically independent of the states, yet extraction of the states can still be performed "faithfully."

Lemma 4. *$\langle \mathcal{C}, \mathcal{R} \rangle$ is non-malleable w.r.t. every* synchronous *adversary.*

Proof. Suppose that the lemma is not true, and there exists a synchronous man-in-the-middle adversary M, an efficient distinguisher D, and a polynomial p such that for infinitely many values of λ, there exist strings (m, m') and a tag id, such that $|\delta_\lambda - \delta'_\lambda| \geq 1/p(n)$ where $\delta_\lambda := \Pr[D(\mathbf{MIM}_{\lambda,\mathsf{id},z}(m)) = 1]$, $\delta'_\lambda := \Pr[D(\mathbf{MIM}_{\lambda,\mathsf{id},z}(m')) = 1]$ where variable $\mathbf{MIM}_{\lambda,\mathsf{id},z}$ is defined w.r.t. M (and $(\lambda, \mathsf{id}, z)$ plays the role of auxiliary input, specified here for emphasis).

Fix any such $\lambda, m, m', \mathsf{id}, z$. Recall that $\mathbf{MIM}_{\lambda,\mathsf{id},z}$ outputs a pair of the form $(\widetilde{\alpha}, \mathsf{VIEW})$. Define ϵ_λ to be the probability that distribution $\mathbf{MIM}_{\lambda,\mathsf{id},z}(m)$ outputs a VIEW that is accepting on right, i.e., M makes a

[2] In more detail, recall that Π_{FS} consists of λ parallel repetitions of [Blu86, LS90], each of which requires committing a cycle as a $q \times q$ matrix. We view $\rho_{\mathsf{FS}} = \{\rho_{i,j,k}\}$ where $\rho_{i,j,k}$ is of length 3λ and used to commit to (i, j)-th entry of the the matrix in k-th repetition using the function $\mathsf{com}_{\rho_{i,j,k}}$, for every $i, j \in [q], k \in [\lambda]$. Also note that the size of x, and hence the value of q is a fixed and a-priori known polynomial in (λ, n, t, ℓ).

successful commitment in the right session of VIEW. Define ϵ'_λ analogously w.r.t. m'. By computational hiding of our scheme, $\epsilon_\lambda, \epsilon'_\lambda$ are negligibly close.

We now design a sequence of hybrid experiments where the first hybrid $\mathcal{H}_0$ corresponds to sampling from the distribution $\mathbf{MIM}_{\lambda,\mathsf{id},z}(m)$ and the last hybrid employs properties of non-malleable codes to sample a computationally indistinguishable output without knowing m.

$\mathcal{H}_0$: This hybrid is identical to the experiment $\mathbf{MIM}_{\lambda,\mathsf{id},z}(m)$. Specifically, it incorporates machines M and D internally, and interacts with M by committing m with identity id and simultaneously receiving a commitment from M w.r.t. some identity $\widetilde{\mathsf{id}}$. Let VIEW denote the view of M at the end of these executions. If VIEW is accepting on right, compute the (unique) value $\widetilde{m}$ committed by M (by running in exponential time) and return the output of $D(\widetilde{m}, \mathsf{VIEW})$.

By construction, VIEW accepts on right with probability ϵ_λ, and $\mathcal{H}_0$ outputs 1 with probability δ_λ.

$\mathcal{H}_1$: This hybrid is identical to $\mathcal{H}_0$ except that it does not extract $\widetilde{m}$ in exponential time; instead it applies the "proof-of-knowledge" extractor to stage-3. We provide the (straightforward) details for future reference. $\mathcal{H}_1$ proceeds as follows:

1. Proceed identically to $\mathcal{H}_0$ until the second messages of stage-3 on both sides $(\mathsf{zk}'_3, \widetilde{\mathsf{zk}}'_3)$ have been sent.[3] Let st denote the state of $\mathcal{H}_1$ up to this point.
2. Continue the execution from st identically to $\mathcal{H}_0$ to finish the last round. Let VIEW be M's view. If VIEW is rejecting on right, or $\widetilde{\mathsf{id}} = \mathsf{id}$, output $D(\perp, \mathsf{VIEW})$. Otherwise, let $\widetilde{x}$ be the statement proven by M in stage-3 on right. Also let $\widetilde{\mathsf{R}}$ be the "second state" in the last message on right.
3. Define a "prover" machine $P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}$ for proof system Π_{FS} as follows. $P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}$ is the machine $\mathcal{H}_0$ initialized to state st with hardwired values (L, v_1, m) and sufficient randomness ζ; it proceeds identically to $\mathcal{H}_0$ from state st but forwards the stage-3 messages on right to an outside verifier. It simulates the left execution for M internally, by computing the last message as follows: sample a random R so that it is consistent with L and $m\|\mathsf{id}$, i.e., $\mathsf{Dec}(\mathsf{L}, \mathsf{R}) = m\|\mathsf{id}$, and then use $(\mathsf{L}, v_1, \mathsf{R}, v_2, m)$ to send the last message where v_2 is chosen randomly.
4. Run the knowledge-extractor $\mathsf{Ext}^{P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}}$ to get a witness $\widetilde{w}$ which includes opening of commitment $\widetilde{c}_1$ to a state $\widetilde{\mathsf{L}}$. Ignore the rest of the witness, compute $\widetilde{m} = \mathsf{Dec}(\widetilde{\mathsf{L}}, \widetilde{\mathsf{R}})$, and output $D(\widetilde{m}, \mathsf{VIEW})$.

Observe that the distribution of VIEW is identical in $\mathcal{H}_1, \mathcal{H}_0$, and Ext extracts a valid witness, and hence correct decommitment of $\widetilde{c}_1$, with probability $1 - \mathsf{negl}(\lambda)$. The outputs of $\mathcal{H}_1, \mathcal{H}_0$ are thus statistically close.

The running time of $\mathcal{H}_1$ is expected polynomial as follows: for a state st, let ϵ_{st} denote the probability that $P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}$ proves a statement; then, the expected running time of $\mathcal{H}_1$ is given by
$\sum_{\mathsf{st}} \left(\epsilon_{\mathsf{st}} \cdot \mathrm{E}\left[\text{Run time of } \mathsf{Ext}^{P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}}\right]\right) \cdot \Pr[\mathsf{st}] \;\; = \sum_{\mathsf{st}} \epsilon_{\mathsf{st}} \cdot \frac{\mathsf{poly}(\lambda)}{\epsilon_{\mathsf{st}}} \cdot \Pr[\mathsf{st}] \;\; = \mathsf{poly}(\lambda)$.

$\mathcal{H}_2$. This hybrid is identical to the previous hybrid except that in the extraction step, it extracts from the prover $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$ where m^* is a random message. Note that this means that during extraction, machine $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$ will now sample state R^* that is consistent with L and message $m^*\|\mathsf{id}$ (fresh every time), so that $\mathsf{Dec}(\mathsf{L}, \mathsf{R}^*) = m^*\|\mathsf{id}$.

We prove that $\mathcal{H}_1$ and $\mathcal{H}_2$ are statistically close as follows. The extraction procedure $\mathsf{Ext}^{P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}}$ finds a witness which includes a valid opening of $\widetilde{c}_1$ with high probability. However, since $\widetilde{c}_1$ is statistically binding, and distributed identically in both hybrids, so is the opened state $\widetilde{\mathsf{L}}$. It follows that the two hybrids are statistically close. Next, we prove that:

[3] Since M is synchronous, these messages are sent before the last rounds of both protocols start.

Claim 13. *$\mathcal{H}_2$ runs in expected polynomial time.*

As noted earlier, the proof of this claim does not follow from standard averaging argument. This is because the success probabilities of $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$ and $P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}$ could be different. We therefore adopt a different strategy by considering several hybrids to move to a statistically, in fact perfectly, hiding transcript/view and then revert back. For better readability, we defer the proof of this claim towards the end.

Let us also note that at this point, we have already started to "split off" the extraction procedure for $\widetilde{\mathsf{L}}$ without the help of R. We will now continue to make small changes until we the extraction of two states can be completely separated from each other.

$\mathcal{H}_3$. This hybrid is identical to the previous hybrid, except that it uses the simulator ZKSim to complete stage 1(e), instead of the real prover algorithm. All other steps are performed as in $\mathcal{H}_2$, using honest values: i.e., the hybrid defines st after stage-2 and completes stages 3 and 4 honestly. If the view is accepting, it performs extraction as before (using $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$).

It is straightforward to see that $\mathcal{H}_2$ runs in expected polynomial time. This is because M is *synchronous* and hence the simulation and extraction steps are not intertwined. Furthermore, due to the zero-knowledge of Π_{FS}, the outputs of $\mathcal{H}_2, \mathcal{H}_1$ are computationally indistinguishable.

$\mathcal{H}_4$: This hybrid is identical to $\mathcal{H}_3$ except that it sets ρ to be of the special form which allows equivocation. More specifically, it selects $N = 2\ell + \lambda q^2$ pairs of seeds of $\{(r_0^i, r_1^i)\}_{i=1}^N$ and sets $\rho := R_1 \| \ldots \| R_N$ where $R_i = \mathsf{prg}(r_0^i) \oplus \mathsf{prg}(r_1^i)$ for all $i \in [N]$. $\mathcal{H}_3$ performs all other steps exactly as in $\mathcal{H}_3$.

Clearly, $\mathcal{H}_3$ runs in expected polynomial time. Its output is computationally indistinguishable from that of $\mathcal{H}_2$ due to the pseudorandomness of prg.

Remark. At this stage, the commitments used in $\mathcal{H}_4$ (for stages 2, 3, and 4 of the protocol on left) are in fact perfectly hiding and independent of the underlying message.

$\mathcal{H}_5$: This hybrid is actually identical to $\mathcal{H}_3$, but we describe it differently using the equivocality property of com. Roughly speaking, the hybrid will simply sample a transcript randomly without knowing the message or the states, but then rely on the equivocality to later find randomness consistent with actual states, to compute the last message. We describe this hybrid in detail, since it will serve as the basis for describing our tampering functions in the next hybrid.

$\mathcal{H}_5$ proceeds as follows:

1. Complete stage-1 exactly as in $\mathcal{H}_3$ using ZKSim to set up $\rho = R_1 \| \ldots \| R_N$ where $R_i = \mathsf{prg}(r_0^i) \oplus \mathsf{prg}(r_1^i)$ for all $i \in [N]$.
2. Complete stage-2 and first two messages of stage-3 by sending *equivocal* commitment strings. Specifically, set $c_1 := \mathsf{prg}(r_0^1) \| \ldots \| \mathsf{prg}(r_0^\ell)$, $\mathsf{zk}_2' := \mathsf{prg}(r_0^{\ell+1}) \| \ldots \| \mathsf{prg}(r_0^{\ell+\lambda q^2})$, and $c_2 := \mathsf{prg}(r_0^{\ell+\lambda q^2+1}) \| \ldots \| \mathsf{prg}(r_0^N)$. Let st denote the state of the hybrid at this point. Continue the execution to also complete the third message of stage-3.
3. Complete the last message zk_4' by using appropriate seeds, i.e., obtain $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m \| \mathsf{id})$, and define randomness v_1, v_2 based on the bits of L, R, namely, $v_1 := (r_0^{\mathsf{L}[1]} \| \ldots \| r_0^{\mathsf{L}[\ell]})$, and v_2 likewise. This defines the witness $w = (\mathsf{L}, \mathsf{R}, v_1, v_2, m)$ for statement x. Next, choose λ random cycles $C := (C_1, \ldots, C_\lambda)$ and define randomness u so that zk_2' is a commitment to C using u w.r.t. string ρ.[4] Complete the last message, zk_4', using (w, C, u).
4. Complete stage-4 by sending v_2 as computed above. Let VIEW be M's view at this point. If VIEW rejects on right, output $D(\bot, \mathsf{VIEW})$.

[4] Informally, for every bit $b_{i,j,k} = C_k[i,j]$, define $u_{i,j,k} = r_{b_{i,j,k}}^{i,j,k}$ so that u is the collection of all $u_{i,j,k}$.

5. Otherwise, proceed as before: extract a witness from $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,(C,u,\zeta')}$ where $\zeta = (C, u, \zeta')$ acts as the randomness. The extracted witness contains $\widetilde{\mathsf{L}}$ and VIEW contains $\widetilde{\mathsf{R}}$ in the last message; output $D(\widetilde{m}, \mathsf{VIEW})$ where m is obtained by decoding $(\widetilde{\mathsf{L}}, \widetilde{\mathsf{R}})$

Using step by step verification, it is easy to see that $(\widetilde{m}, \mathsf{VIEW})$ is distributed identically in both $\mathcal{H}_5, \mathcal{H}_4$. It is also easy to check that $\mathcal{H}_5$ runs in expected polynomial time.

Remark: Note that only step 3 above needs access to the message m for completing the last zero-knowledge message zk'_4.

$\mathcal{H}_6$: In this hybrid, instead of using m to compute the last zero-knowledge message zk'_4, we use states of a random message. All other steps are still performed using correct states (L, R). Formally, $\mathcal{H}_6$ starts by sampling two encodings: $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(m \| \mathsf{id})$ and $(\widehat{\mathsf{L}}, \widehat{\mathsf{R}}) \leftarrow \mathsf{Enc}(\widehat{m} \| \mathsf{id})$ where $\widehat{m}$ is chosen randomly. It then proceeds as follows:

1. Proceed identically to $\mathcal{H}_4$ up to the first three messages of stage-3
2. Obtain randomness $\widehat{v}_1, \widehat{v}_2$ w.r.t. $\widehat{\mathsf{L}}, \widehat{\mathsf{R}}$ as before to define the witness $\widehat{w} = (\widehat{\mathsf{L}}, \widehat{v}_1, \widehat{\mathsf{R}}, \widehat{v}_2, \widehat{m})$. Compute message zk'_4 using $(\widehat{w}, C, u)$ as in the previous hybrid and finish stage-3.
3. Finish stage-4 using the *correct* states as in the previous hybrid. I.e., find randomness v_1, v_2 w.r.t. L, R and send v_2 in stage-4. Let VIEW^* denote M's view at this point.
4. If VIEW^* is accepting on right, extract $\widetilde{\mathsf{L}}$ from $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,(C,u,\zeta')}$ as before and use D to return the output.

The only difference between two hybrids is in the computation of zk'_4. However, zk'_4 is independent of the underlying "witness" due to the perfect-hiding property of underlying equivocal commitments. Therefore $\mathcal{H}_6, \mathcal{H}_5$ are statistically close.

$\mathcal{H}_7$: In this hybrid we first define a family of tampering functions which use $\mathcal{H}_6$ internally. Observe that $\mathcal{H}_6$ does not use (L, R) until the last message (which consists of zk'_4, and v_2). Furthermore, it computes $\widetilde{\mathsf{R}}$ without the knowledge of L and, likewise, extracts $\widetilde{\mathsf{L}}$ without the knowledge of R. Therefore, we already have the ability to perform extraction (or "tampering") individually. With this observation, we define the following family $\mathcal{F}_{\mathsf{split}}$ of tampering functions.

Let Φ denote the set of all possible random tapes for hybrid $\mathcal{H}_6$. Define $F = \{(f_\phi, g_\phi)\}_{\phi \in \Phi}$ where f_ϕ, g_ϕ are defined below (we define g_ϕ first):

1. **Function** $g_\phi : \{0,1\}^\ell \to \{0,1\}^\ell$ takes as input a state R. It proceeds exactly as $\mathcal{H}_6$ with randomness ϕ up to the completion of stage-4 where VIEW^* is obtained. Let $\widetilde{\mathsf{R}}$ denote the state opened by M in stage-4 on right. Output $\widetilde{\mathsf{R}}$.
2. **Function** $f_\phi : \{0,1\}^\ell \to \{0,1\}^\ell$ takes as input a state s_1. It proceeds exactly as $\mathcal{H}_6$ with randomness ϕ up to the completion of stage-3. It does not complete stage-4; instead, it directly runs the extractor exactly as $\mathcal{H}_6$, and outputs the first component of the extracted witness, denoted $\widetilde{\mathsf{L}}$.

We now describe $\mathcal{H}_7$ using functions in F. Specifically, $\mathcal{H}_7$, samples states $(\mathsf{L}, R) \leftarrow \mathsf{Enc}(m \| \mathsf{id})$, and proceeds identically to $\mathcal{H}_6$ with fresh randomness ϕ and state R to obtain VIEW^*. However, $\mathcal{H}_6$ does not run the extractor directly. Instead, if VIEW^* is accepting, it computes

$$\widetilde{\mathsf{L}} = f_\phi(\mathsf{L}), \quad \widetilde{\mathsf{R}} = g_\phi(\mathsf{R}), \quad \widetilde{m} \| \widetilde{\mathsf{id}} = \mathsf{Dec}(\widetilde{\mathsf{L}}, \widetilde{\mathsf{R}}),$$

and outputs $D(\widetilde{m}, \mathsf{VIEW}^*)$ if $\widetilde{\mathsf{id}} \neq \mathsf{id}$. Otherwise it outputs $D(\perp, \mathsf{VIEW}^*)$.

By construction, $\mathcal{H}_6, \mathcal{H}_7$ are identical. Let $\delta^{(7)}_\lambda$ denote the probability that $\mathcal{H}_7$ outputs 1. We have that $|\delta^{(7)}_\lambda - \delta_\lambda| \leq \mathsf{negl}(\lambda)$. We claim that $\delta^{(7)}_\lambda$ is independent of m. We prove this by obtaining an explicit expression for $\delta^{(7)}_\lambda$ using the non-malleability of the code.

Let us fix some notation. Observe that fixing a random tape ϕ, completely fixes variable VIEW^* sampled above. Therefore, we will write VIEW^*_ϕ to refer to VIEW^* corresponding to ϕ. Let $\mathbb{A}$ denote the set of VIEW^* that are "accepting" on right. Note that we have already argued that VIEW^* is independent of the message m. Therefore, $\Pr[\phi \in \mathbb{A}]$ is also independent of m. Also observe that sampling of $\widetilde{m}\|\widetilde{\mathsf{id}}$ in $\mathcal{H}_7$ w.r.t. tape ϕ is identical to sampling from the "tampered distribution" $\mathsf{T}_{m\|\mathsf{id},f_\phi,g_\phi}$ and this distribution is simulatable, i.e., there exists negligible function negl_ϕ and a simulator $\mathsf{Sim}_{f_\phi,g_\phi}$ such that $\Delta(\mathsf{T}_{m\|\mathsf{id},f_\phi,g_\phi}, \mathsf{Copy}^{(m\|\mathsf{id})}_{\mathsf{Sim}_{f_\phi,g_\phi}}) = \mathsf{negl}_\phi(\lambda)$ where $\mathsf{Copy}^{(m\|\mathsf{id})}_{\mathsf{Sim}_{f_\phi,g_\phi}}$ samples $\widetilde{m} \leftarrow \mathsf{Sim}_{f_\phi,g_\phi}$ and outputs $m\|\mathsf{id}$ if $\widetilde{m} = \mathsf{same}$ and $\widetilde{m}$ otherwise.

We have,

$$\begin{aligned}
\delta^{(7)}_\lambda &= \sum_{\phi\in\mathbb{A}} \Pr\left[D(\widetilde{m},\mathsf{VIEW}^*_\phi)=1 \,\middle|\, \begin{array}{c}\widetilde{m}\|\widetilde{\mathsf{id}} \leftarrow \mathsf{T}_{m\|\mathsf{id},f_\phi,g_\phi}\\ \text{if } \widetilde{\mathsf{id}} = \mathsf{id} \text{ set } \widetilde{m} := \perp\end{array}\right]\cdot\Pr[\phi] + \underbrace{\sum_{\phi\notin\mathbb{A}} \Pr\left[D(\perp,\mathsf{VIEW}^*_\phi)=1\right]\cdot\Pr[\phi]}_{=p \ \text{(independent of } m)}\\
&= \sum_{\phi\in\mathbb{A}} \Pr\left[D(\widetilde{m},\mathsf{VIEW}^*_\phi)=1 \,\middle|\, \begin{array}{c}\widetilde{m}\|\widetilde{\mathsf{id}} \leftarrow \mathsf{Copy}^{(m\|\mathsf{id})}_{\mathsf{Sim}_{f_\phi,g_\phi}}\\ \text{if } \widetilde{\mathsf{id}} = \mathsf{id} \text{ set } \widetilde{m} := \perp\end{array}\right]\cdot\Pr[\phi] + \sum_{\phi\in\mathbb{A}} \pm\mathsf{negl}_\phi\cdot\Pr[\phi] + p\\
&= \sum_{\phi\in\mathbb{A}} \Pr\left[D(\widetilde{m},\mathsf{VIEW}^*_\phi)=1 \,\middle|\, \begin{array}{c}\beta \leftarrow \mathsf{Sim}_{f_\phi,g_\phi}\\ \text{if } \beta = \mathsf{same} \text{ set } \widetilde{m} := \perp\\ \text{else } \widetilde{m} := \beta[1\ldots n]\end{array}\right]\cdot\Pr[\phi] + \underbrace{\sum_{\phi\in\mathbb{A}} \pm\mathsf{negl}_\phi\cdot\Pr[\phi]}_{=\text{negligible}} + p\\
&= \text{value independent of } m.
\end{aligned}$$

Now consider the case of m' and the variable δ'_λ w.r.t. $\mathbf{MIM}_{\lambda,\mathsf{id},z}(m')$. Using the same argument as above, we can conclude that $\mathbf{MIM}_{\lambda,\mathsf{id},z}(m')$ is also computationally indistinguishable from hybrid $\mathcal{H}_7$. Therefore, $|\delta'_\lambda - \delta^{(7)}_\lambda| \le \mathsf{negl}(\lambda)$ and hence $|\delta_\lambda - \delta'_\lambda| \le 4\mathsf{negl}(\lambda)$ which is a contradiction. □

Proof of claim 13. To prove that $\mathcal{H}_2$ runs in expected polynomial time, we have to consider a series of hybrid experiments to obtain statistically independent transcripts. At this point, we can calculate the running time in a standard manner since there would be no difference in success probabilities when we switch m to m^* in the prover machine. We now provide a sketch of the hybrids (since they are very similar to our previous hybrids).

Consider the following sequence of hybrids:

$\mathcal{H}_{1:1}$. Identical to $\mathcal{H}_1$ except that it uses ZKSim to complete stage 1(e). Note that $\mathcal{H}_{1:1}$ is expected polynomial time; furthermore, the extracted witness and in particular its first component $\widetilde{\mathsf{L}}$ is computationally indistinguishable from $\mathcal{H}_1$. (Note that M is still synchronous)

$\mathcal{H}_{1:2}$. Identical to $\mathcal{H}_{1:1}$ except that we switch the strings ρ to be of the special form. (This is done in the same manner as hybrid $\mathcal{H}_4$ and we rely on the security of prg). Note that at this point the transcripts are independent of the underlying witness. (See $\mathcal{H}_5$ for a detailed explanation)

$\mathcal{H}_{1:3}$. Identical to $\mathcal{H}_{1:2}$ except that the extraction is now performed using the prover machine $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$ where m^* is a random message.

We claim that $\mathcal{H}_{1:3}$ runs in expected polynomial time. Let ϵ_{st} denote the success probability of prover $P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}$ (in the previous hybrid). Observe that the two hybrids define the same distribution on states st. The only difference between $P^{\mathsf{st}}_{\mathsf{L},v_1,m,\zeta}$ and $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$ is that they compute the last zero-knowledge message zk'_4 (of stage-3) using a different witness. However, since the protocol transcript hides the witness perfectly, we have that for every st, $\epsilon_{\mathsf{st}} = \epsilon^*_{\mathsf{st}}$ where ϵ^*_{st} is the success probability of $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$. The running time of

$\mathcal{H}_{1:3}$ is thus given by $\sum_{\mathsf{st}} \left(\epsilon_{\mathsf{st}} \cdot \mathrm{E}\left[\text{Run time of } \mathsf{Ext}^{P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}} \right] \right) \cdot \Pr[\mathsf{st}] \quad = \sum_{\mathsf{st}} \epsilon_{\mathsf{st}} \cdot \frac{\mathsf{poly}(\lambda)}{\epsilon^*_{\mathsf{st}}} \cdot \Pr[\mathsf{st}] \quad = \mathsf{poly}(\lambda)$.

Note that we have already switched to the prover $P^{\mathsf{st}}_{\mathsf{L},v_1,m^*,\zeta}$. Now we simply reverse the hybrids: in $\mathcal{H}_{1:4}$ we switch back to using a normal string ρ, and in $\mathcal{H}_{1:5}$ we start using the real prover algorithm instead of ZKSim (details are straightforward). Observe that $\mathcal{H}_{1:4} = \mathcal{H}_2$, and hence the claim. □

8.2 Non-malleability against Asynchronous Strategies

Lemma 5. *$\langle \mathcal{C}, \mathcal{R} \rangle$ is non-malleable w.r.t. every* asynchronous *adversary.*

Proof. We now consider an asynchronous adversary who follows any strategy different from the synchronous. We perform a case by case analysis, depending upon the schedule of the adversary, and show that each case either reduces to the synchronous case, or it is trivially non-malleable due to the hiding and extractability properties of the scheme. Specifically, there are three representative cases, as described below (and depicted pictorially in figure 5):[5]

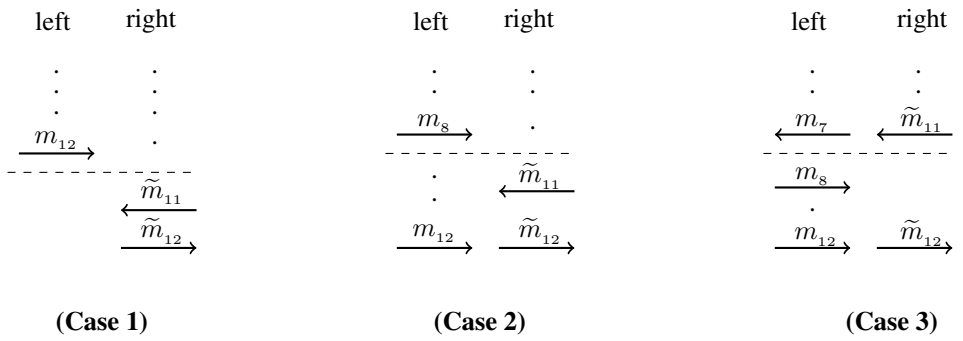

Figure 5: Representative schedules for an asynchronous adversary for proof of lemma 5.

Case 1: The last two messages on right appear after all messages on left.

This case is trivially non-malleable due to the hiding of left side commitment since $\widetilde{s}_1$ on right can be extracted without rewinding the left protocol to obtain the committed message $\widetilde{m}$.

Note: If the last message on right, $\widetilde{m}_{12}$, appears after the left protocol ends, then also the protocol is non-malleable since M has committed to his message before the message on left has been fixed. Therefore, in the following cases, we only need to consider the case when $\widetilde{m}_{12}$ is synchronous with m_{12}.

Case 2: The last two messages on right appear after stage-1 Π_{FS} on left.

This case is handled the same way as the synchronous case. Specifically, since the extraction on right still does not interfere with simulation on left, the proof for this case goes through exactly as for the synchronous case.

Case 3: The non-malleability for this case also follows from hiding of left side commitment, because the last two message on right "completely contain" the stage-3 and 4 of left protocol. Therefore, rewinding on right will completely rewind this stage (past the first message) and hence computational hiding of left protocol is maintained. We provide a proof sketch.

[5]Before reading the cases/figure, let us note that the messages of the left protocol are denoted by $m_1, \ldots, m_{12}$ where m_8 denotes the end of stage-1 as well as 2 (since both messages are sent together), and m_7 is the challenge for stage-1 Π_{FS} on left. The corresponding messages on right are denoted by $\widetilde{m}_1, \ldots, \widetilde{m}_{12}$.

As in proof of lemma 4, fix λ, M, D, z and values $\epsilon_\lambda, \delta_\lambda, \delta'_\lambda$ for one of infinitely many λ such that ϵ_λ (the probability of a successful commitment on right) is a inverse polynomial so is the difference between $\delta_\lambda, \delta'_\lambda$. We perform extraction for this λ and violate hiding of stage-4 commitment (which is actually a ZK protocol but can be viewed as a commitment to m (or m') through commitment to its first state and the second state given in plain) for this λ.

Consider a machine D' which internally incorporates M and D and has been initialized up to the point where it awaits $\widetilde{m}_{11}$ on right (see fig. 5). It gets a commitment from outside (to either m or m') and forwards it internally to M. If M fails on right, D' outputs a random bit as the guess. Otherwise, D' proceeds for extraction (from the last two messages)—while completing the left side execution using a *random* message whenever needed.

Let ϵ^*_λ denote the probability that M makes a successful commitment on right when the left side is completed with a random message m^*. Clearly, $\epsilon_\lambda, \epsilon^*_\lambda$ are negligibly close. Note that by our assumption, ϵ_λ is an inverse polynomial, and therefore so must be ϵ^*_λ. After extraction, D' outputs whatever D would.

The expected running time of D' is dominated by the term $\epsilon_\lambda \cdot \frac{\mathsf{poly}(\lambda)}{\epsilon^*_\lambda}$; which is bounded by a polynomial due to our assumptions on $\epsilon_\lambda, \epsilon^*_\lambda$. It is clear that in this case the value is extracted with probability $1 - \mathsf{negl}(\lambda)$ and the output is the same as that of D in the real execution with m (resp. m'). Therefore D' breaks hiding if D breaks non-malleability.

□

9 Acknowledgements

We would like to thank Divesh Aggarwal, Anand Degwekar, Yevgeniy Dodis and Antigoni Polychroniadou for helpful discussions and insights.

References

[AAG+16] Divesh Aggarwal, Shashank Agrawal, Divya Gupta, Hemanta Maji, Omkant Pandey, and Manoj Prabhakaran. Optimal computational split-state non-malleable codes. In *TCC*, 2016.

[ADKO15] Divesh Aggarwal, Yevgeniy Dodis, Tomasz Kazana, and Maciej Obremski. Non-malleable reductions and applications. In *Proceedings of the Forty-Seventh Annual ACM on Symposium on Theory of Computing, STOC 2015, Portland, OR, USA, June 14-17, 2015*, pages 459–468, 2015.

[ADL14] Divesh Aggarwal, Yevgeniy Dodis, and Shachar Lovett. Non-malleable codes from additive combinatorics. In *Symposium on Theory of Computing, STOC 2014, New York, NY, USA, May 31 - June 03, 2014*, pages 774–783, 2014.

[Agg] Divesh Aggarwal. Personal communication, 10/30/2015.

[Bar02] Boaz Barak. Constant-Round Coin-Tossing with a Man in the Middle or Realizing the Shared Random String Model. In *Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science*, FOCS '02, pages 345–355, 2002.

[BGR+15] Hai Brenner, Vipul Goyal, Silas Richelson, Alon Rosen, and Margarita Vald. Fast non-malleable commitments. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, CO, USA, October 12-6, 2015*, pages 1048–1057, 2015.

[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.

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

[CDMW08] Seung Geol Choi, Dana Dachman-Soled, Tal Malkin, and Hoeteck Wee. Black-box construction of a non-malleable encryption scheme from any semantically secure one. In *Theory of Cryptography, Fifth Theory of Cryptography Conference, TCC 2008, New York, USA, March 19-21, 2008.*, volume 4948 of *Lecture Notes in Computer Science*, pages 427–444. Springer, 2008.

[CGL15] Eshan Chattopadhyay, Vipul Goyal, and Xin Li. Non-malleable extractors and codes, with their many tampered extensions. *CoRR*, abs/1505.00107, 2015.

[CGMO09] Nishanth Chandran, Vipul Goyal, Ryan Moriarty, and Rafail Ostrovsky. Position based cryptography. In Shai Halevi, editor, *CRYPTO*, volume 5677 of *Lecture Notes in Computer Science*, pages 391–407. Springer, 2009.

[CLOS02] Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, STOC '02, pages 494–503, 2002.

[CZ14] Eshan Chattopadhyay and David Zuckerman. Non-malleable codes against constant split-state tampering. In *55th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2014, Philadelphia, PA, USA, October 18-21, 2014*, pages 306–315, 2014.

[DDN91] Danny Dolev, Cynthia Dwork, and Moni Naor. Non-Malleable Cryptography (Extended Abstract). In *Proceedings of the 23rd Annual ACM Symposium on Theory of Computing*, STOC '91, pages 542–552, 1991.

[DKO13] Stefan Dziembowski, Tomasz Kazana, and Maciej Obremski. Non-malleable codes from two-source extractors. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II*, volume 8043 of *Lecture Notes in Computer Science*, pages 239–257. Springer, 2013.

[DPW10] Stefan Dziembowski, Krzysztof Pietrzak, and Daniel Wichs. Non-malleable codes. In *Innovations in Computer Science - ICS 2010, Tsinghua University, Beijing, China, January 5-7, 2010. Proceedings*, pages 434–452, 2010.

[FS89] Uriel Feige and Adi Shamir. Zero knowledge proofs of knowledge in two rounds. In *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*, pages 526–544, 1989.

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *STOC*, pages 416–426. ACM, 1990.

[GK96] Oded Goldreich and Hugo Krawczyk. On the composition of zero-knowledge proof systems. *SIAM J. Comput.*, 25(1):169–192, 1996.

[GLOV12] Vipul Goyal, Chen-Kuei Lee, Rafail Ostrovsky, and Ivan Visconti. Constructing non-malleable commitments: A black-box approach. In *FOCS*, pages 51–60. IEEE Computer Society, 2012.

[GMW87] Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*, pages 218–229, 1987.

[Goy11] Vipul Goyal. Constant Round Non-malleable Protocols Using One-way Functions. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing*, STOC '11, pages 695–704. ACM, 2011.

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

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A Pseudorandom Generator from any One-way Function. *SIAM J. Comput.*, 28(4):1364–1396, 1999.

[IKOS07] Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Zero-knowledge from Secure Multiparty Computation. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*, STOC '07, pages 21–30, 2007.

[Kiy14] Susumu Kiyoshima. Round-efficient black-box construction of composable multi-party computation. In Juan A. Garay and Rosario Gennaro, editors, *Advances in Cryptology - CRYPTO 2014 - 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part II*, volume 8617 of *Lecture Notes in Computer Science*, pages 351–368. Springer, 2014.

[KMO14] Susumu Kiyoshima, Yoshifumi Manabe, and Tatsuaki Okamoto. Constant-round black-box construction of composable multi-party computation protocol. In Yehuda Lindell, editor, *Theory of Cryptography - 11th Theory of Cryptography Conference, TCC 2014, San Diego, CA, USA, February 24-26, 2014. Proceedings*, volume 8349 of *Lecture Notes in Computer Science*, pages 343–367. Springer, 2014.

[LP09] Huijia Lin and Rafael Pass. Non-malleability Amplification. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing*, STOC '09, pages 189–198, 2009.

[LP11] Huijia Lin and Rafael Pass. Constant-round Non-malleable Commitments from Any One-way Function. In *Proceedings of the 43rd Annual ACM Symposium on Theory of Computing*, STOC '11, pages 705–714, 2011.

[LP12] Huijia Lin and Rafael Pass. Black-box constructions of composable protocols without set-up. In Reihaneh Safavi-Naini and Ran Canetti, editors, *Advances in Cryptology - CRYPTO 2012 - 32nd Annual Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2012. Proceedings*, volume 7417 of *Lecture Notes in Computer Science*, pages 461–478. Springer, 2012.

[LPV08] Huijia Lin, Rafael Pass, and Muthuramakrishnan Venkitasubramaniam. Concurrent Non-malleable Commitments from Any One-Way Function. In *Theory of Cryptography, 5th Theory of Cryptography Conference, TCC 2008*, pages 571–588, 2008.

[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.

[LS90] Dror Lapidot and Adi Shamir. Publicly verifiable non-interactive zero-knowledge proofs. In *CRYPTO*, pages 353–365, 1990.

[Nao91] Moni Naor. Bit Commitment Using Pseudorandomness. *J. Cryptology*, 4(2):151–158, 1991.

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

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

[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.

[PR05b] Rafael Pass and Alon Rosen. New and improved constructions of non-malleable cryptographic protocols. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, STOC '05, pages 533–542, 2005.

[PW10] Rafael Pass and Hoeteck Wee. Constant-Round Non-malleable Commitments from Sub-exponential One-Way Functions. In *Advances in Cryptology — EUROCRYPT '10*, pages 638–655, 2010.

[Wee10] Hoeteck Wee. Black-Box, Round-Efficient Secure Computation via Non-malleability Amplification. In *Proceedings of the 51th Annual IEEE Symposium on Foundations of Computer Science*, pages 531–540, 2010.

A Failed Attempts: Selective $\perp$ Attacks

We describe a few "false starts" which exemplify the difficulty in attaining three round non-malleable commitment. We restrict our attention to mauling attacks which could be ruled out with a zero-knowledge proof of correctness as other attacks have already been addressed in prior work. We leave out the identities for this discussion. To start with (and disregarding the lower bound of [Pas13]), consider the non-interactive "encode-then-encrypt" protocol where $\mathcal{C}$ commits to v by drawing $(\mathsf{L}, \mathsf{R}) \leftarrow \mathsf{Enc}(v)$ and sending $\mathrm{Com}(\mathsf{L}), \mathrm{Com}(\mathsf{R})$ to $\mathcal{R}$. One might hope that committing separately to the states would allow us to argue that in any mauling attack, the PPT M must treat the states individually and so non-malleability would follow from that of the code. This intuition does not work as Com possesses no non-malleability guarantees, and so we cannot rule out the possibility that M will maul (L, R) jointly to $(\tilde{\mathsf{L}}, \tilde{\mathsf{R}})$, an encoding of say $v+1$.

Consider instead the scheme where $\mathcal{C}$ sends $\mathrm{Com}(\mathsf{L})$ in the first round, $\mathcal{R}$ checks back with an acknowledgement message, then $\mathcal{C}$ sends R. This protocol seems like it should be non-malleable: M is forced to maul L to $\tilde{\mathsf{L}}$ before he sees R, and the hiding of Com should ensure that M is mauling R without knowledge of L. However, this intuition turns out not to be sound. Given any non-malleable code $(\mathsf{Enc}_0, \mathsf{Dec}_0)$, consider the pathological non-malleable code where $\mathsf{Enc}(m)$ samples $(\mathsf{L}_0, \mathsf{R}_0) \leftarrow \mathsf{Enc}_0(m)$ and outputs $\mathsf{L} = (\mathsf{L}_0, \sigma)$ and $\mathsf{R} = (\mathsf{R}_0, z)$ where $\sigma, z \leftarrow \$$; $\mathsf{Dec}(\mathsf{L}, \mathsf{R}) = \mathsf{Dec}_0(\mathsf{L}_0, \mathsf{R}_0)$ unless z is a commitment to the message (L_0, σ) using randomness σ, in which case $\mathsf{Dec}(\mathsf{L}, \mathsf{R}) = \mathsf{Dec}_0(\mathsf{L}_0, \mathsf{R}_0)+1$. It is easy to show that $(\mathsf{Enc}, \mathsf{Dec})$ is non-malleable if $(\mathsf{Enc}_0, \mathsf{Dec}_0)$ is. But now if the above protocol is instantiated on top of such a code, M can maul $\mathrm{Com}\big((\mathsf{L}_0, \sigma); \tau\big)$ to $\tilde{z} = \mathrm{Com}\big((\mathsf{L}_0, \tilde{\sigma}); \tilde{\sigma}\big)$ (this mauling is happening under the Com so M needn't know $\tilde{\sigma}$ to perform such an attack), then M can maul (R_0, z) to $(\mathsf{R}_0, \tilde{z})$. M has successfully committed to $v+1$. It is important to force M to demonstrate knowledge of L in the third round.

Finally, consider a version of our protocol but without any non-malleable codes. In this scheme $\mathcal{C}$ sends $\mathrm{Com}(v)$ and $\mathrm{Com}(r)$ for a random $r \leftarrow \mathbb{Z}_q$, $\mathcal{R}$ sends a random $\alpha \leftarrow \mathbb{Z}_q$ and $\mathcal{C}$ answers with $r\alpha + v$. This is the main subprotocol from the [GRRV14] scheme but with the zero-knowledge proof of consistency removed. Also forget about the copying attack available to M in this discussion as this can be prevented using non-malleability amplification. Still, M has the following selective $\perp$ attack available. Upon receiving $\mathrm{Com}(v)$, $\mathrm{Com}(r)$ M produces $\mathrm{Com}(\tilde{v})$, $\mathrm{Com}(v)$ on the right, where $\tilde{v}$ is the (unrelated) value to which he wants to commit. He disregards the remainder of the left protocol. Upon receiving $\tilde{\alpha}$, M answers with $\tilde{\alpha} + \tilde{v}$. If $v = 1$, then M has

successfully committed to $\tilde{v}$; if $v \neq 1$, he has committed to $\perp$. Such selective $\perp$ attacks seem to be available to M in any protocol in which $\mathcal{C}$'s message is fully specified in the first round.