Two-Round and Non-interactive Concurrent Non-Malleable Commitments from Time-Lock Puzzles

Huijia Lin* Rafael Pass† Pratik Soni*

April 21, 2017

Abstract

Non-malleable commitments are a fundamental cryptographic tool for preventing against (concurrent) man-in-the-middle attacks. Since their invention by Dolev, Dwork, and Naor in 1991, the round-complexity of non-malleable commitments has been extensively studied, leading up to constant-round concurrent non-malleable commitments based only on one-way functions, and even 3-round concurrent non-malleable commitments based on subexponential one-way functions.

But constructions of *two-round*, or *non-interactive*, non-malleable commitments have so far remained elusive; the only known construction relied on a strong and non-falsifiable assumption with a non-malleability flavor. Additionally, a recent result by Pass shows the impossibility of basing two-round non-malleable commitments on falsifiable assumptions using a polynomial-time black-box security reduction.

In this work, we show how to overcome this impossibility, using super-polynomial-time hardness assumptions. Our main result demonstrates the existence of a two-round concurrent non-malleable commitment based on sub-exponential "standard-type" assumptions—notably, assuming the existence of the following primitives (all with subexponential security): (1) non-interactive commitments, (2) ZAPs (i.e., 2-round witness indistinguishable proofs), (3) collision-resistant hash functions, and (4) a "weak" time-lock puzzle.

Primitives (1),(2),(3) can be based on e.g., the discrete log assumption and the RSA assumption. Time-lock puzzles—puzzles that can be solved by "brute-force" in time 2^t, but cannot be solved significantly faster even using parallel computers—were proposed by Rivest, Shamir, and Wagner in 1996, and have been quite extensively studied since; the most popular instantiation relies on the assumption that 2^t repeated squarings mod $N = pq$ require "roughly" 2^t parallel time. Our notion of a "weak" time-lock puzzle, requires only that the puzzle cannot be solved in parallel time 2^{t^ε} (and thus we only need to rely on the relatively mild assumption that there are no *huge* improvements in the parallel complexity of repeated squaring algorithms).

We additionally show that if replacing assumption (2) for a non-interactive witness indistinguishable proof (NIWI), and (3) for a *uniform* collision-resistant hash function, then a *non-interactive* (i.e., one-message) version of our protocol satisfies concurrent non-malleability w.r.t. uniform attackers.

*{rachel.lin,pratik_soni}@cs.ucsb.edu. Supported in part by NSF grants CNS-1528178 and CNS-1514526.

†rafael@cs.cornell.edu. Supported in part by an Alfred P. Sloan Fellowship, a Microsoft New Faculty Fellowship, NSF Awards CNS-1217821 and CCF-1214844, NSF CAREER Award CCF-0746990, AFOSR Award FA9550-08-1-0197, AFOSR YIP Award FA9550-10-1-0093, BSF Grant 2006317, and DARPA and AFRL under contract FA8750-11-2-0211. The views and conclusions contained in this document are those of the authors and should not be interpreted as the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

Contents

1 Introduction

Commitment schemes are one of the most fundamental cryptographic building blocks. Often described as the "digital" analogue of sealed envelopes, commitment schemes enable a *sender* to commit itself to a value while keeping it secret from the *receiver*. This property is called *hiding*. Furthermore, the commitment is *binding*, and thus in a later stage when the commitment is opened, it is guaranteed that the "opening" can yield only a single value determined in the committing stage.

For many applications, however, the most basic security guarantees of commitments are not sufficient. For instance, the basic definition of commitments does not rule out an attack where an adversary, upon seeing a commitment to a specific value v, is able to commit to a related value (say, $v - 1$), even though it does not know the actual value of v. To address this concern, Dolev, Dwork and Naor (DDN) introduced the concept of *non-malleable commitments* [DDN00]. Loosely speaking, a commitment scheme is said to be non-malleable if it is infeasible for an adversary to "maul" a commitment to a value v into a commitment to a related value $\tilde{v}$. The notion of a *concurrent non-malleable commitment* [DDN00, PR05a] further requires non-malleability to hold even if the adversary receives many commitments and can itself produce many commitments.

The first non-malleable commitment protocol was constructed in the original work of [DDN00] in 1991, based on the minimal assumption of one-way functions. The first concurrently secure construction was provided by Pass and Rosen in 2005 [PR05a]. Since then, a central question in the study of non-malleability has been to determine the exact number of communication rounds needed for acheiving (concurrent) non-malleable commitments. Significant progress has been made over the years [Bar02, PR05a, PR05b, LPV08, LP09, PPV08, PW10, Wee10, Goy11, LP11, GLOV12]. The current state-of-the-art is that 4-round concurrent non-malleable commitments can constructed based on one-way functions [COSV16a] and 3-round concurrent non-malleable commitments can be constructed from subexponential-secure injective one-way functions [GRRV14, GPR16, COSV16b].

On the Existence of Two-Round or Non-Interactive Non-malleable Commitments: The situation changes drastically when it comes to two-round or non-interactive (i.e., one-message) protocols: Pandey, Pass and Vaikuntanathan [PPV08] provided a construction of a non-interactive non-malleable commitment based on a new *non-falsifiable* hardness assumption, namely, the existence of an *adaptively-secure injective one-way function*—roughly speaking, a one-way function f that is hard to invert on a random point $y = f(x)$ even if you get access to an inversion oracle that inverts it on every *other* point $y' \neq y$. This assumption is not falsifiable since the inversion oracle cannot be implemented in "real-life"; additionally, note that the assumption also has a strong non-malleability flavor—in particular, the assumption would clearly be false if one could "maul" $y = f(x)$ to e.g., $y' = f(x + 1)$. As such, this construction gives us little insight into whether we can obtain two-round "non-malleability" from "pure scratch" (i.e., from "hardness" alone). Indeed, a recent work by Pass [Pas13] showed that there are some inherent limitations to reducing 2-round non-malleability from falsifiable assumptions. More precisely, Pass shows that if a 2-round non-malleable commitment that can be proven secure using a polyomial-time (or even super-polynomial, but security preserving) black-box reduction R, then the reduction R can itself break the assumption. In particular, this rules out basing 2-round non-malleability (using black-box reduction) on falsifiable polynomial-time hardness assumptions.

Towards overcoming this barrier, a recent work by Goyal, Khurana and Sahai [GKS16] presents a two-message protocol in a stronger "synchronous model" of communication (and achieving only a weaker notion of notion of non-malleability "w.r.t. opening"). In this work, we focus on the standard communication model (and the standard notion of non-malleability) and explore whether

super-polynomial-time hardness assumptions (and using non-security preserving reductions) can be used to overcome this barrier:

Can we have obtain non-interactive or 2-round non-malleable commitment from super-polynomial "standard-type" assumptions?

1.1 Our Results

Our main result demonstrates the existence of a two-round concurrent non-malleable commitment scheme based on sub-exponential "standard-type" assumptions—notably, assuming the existence of the following primitives (all with subexponential security): (1) non-interactive commitments, (2) ZAPs (i.e., 2-round witness indistinguishable proofs) [DN00], (3) collision-resistant hashfunctions, and (4) a "weak" time-lock puzzle [RSW96].

Primitives (1),(2),(3) are all very common used and can be based on e.g., the discrete log assumption and the RSA assumption. Primitive (4) deserves some more discussion: *Time-lock puzzles*—roughly speaking, puzzles that can be solved in "brute-force" in time 2^t, but cannot be solved "significantly faster" even using parallel computers—were proposed by Rivest, Shamir, and Wagner in 1996 [RSW96] (following May's work on time-released cryptography [May93]), and have since been quite extensively in the area of time-released cryptography. A bit more precisely, a $(T(\cdot), B(\cdot))$-*time-lock puzzle* enables a "sender" to efficiently generate a puzzle puz with a solution s and a designated "level" of hardness $t = t(n)$ where n is the security parameter, so that: (i) the puzzle solution can be found in (uniform) time 2^t, but (ii) the puzzle solution cannot be recovered by any $B(n)$-size attacker with (parallel) running-time (i.e., circuit depth) $T = T(t)$ (where $T(t) << t$ is determines the "hardness gap" of the puzzle). Typical applications of time-lock puzzle only require security against polynomial-size attackers, thus it suffices to let $B(\cdot)$ be any slightly super-polynomial function; however, they require the hardness gap to be very small—namely, $T = 2^{\delta t}$ or even $T = \delta 2^t$ (i.e., the problem is inherently "sequential" and the honest puzzle solver is essentially is optimal, even if you have access to parallel computers). In this work, we will need security against subexponential-size attackers, but in contrast, only require the existence of a time-lock puzzle with a relatively "large" hardness gap—we only need the puzzle to be hard to break for time $T = 2^{n^\varepsilon}$ for some constant ε.

Theorem 1 (Main Theorem, Informal)**.** *Let T and B be two arbitrary subexponential functions. Assume the existence of a non-interactive commitments, a ZAP, a family of collision-resistant hash functions, all with subexponential-security, and the existence of a (T, B)-time-lock puzzle. Then, there exists a 2-round concurrent non-malleable commitment.*

The original construction of time-lock puzzles due to Rivest, Shamir, and Wagner [RSW96] is based on the hardness of a very natural strengthening of the factoring problem referred to as the *repeated squaring problem*: given a random RSA-modulus $N = pq$, and a random (or appropriately chosen) element g, compute

$$g^{2^{2^t}} \bmod N$$

Clearly, this can be done using 2^t repeated squarings. The RSW assumption is that this task cannot be significantly sped up, even using parallel resources, unless N can be factored. Given the current state-of the art, the repeated squaring problem appears to be hard for *strongly exponential* parallel-time: $T(t) = \delta 2^t$ (that is, basically, no non-trivial speed-up to repeated squaring is possible); indeed, this strong assumption is typically used in the literature on time-released cryptography (in fact, several significantly stronger versions of this assumption, where additional leakage is given, are

also typically considered—see e.g., the "generalized Blum-Blum-Schub assumption" of Boneh-Naor [BN00].)

Since we only need a "weakly"-secure time-lock puzzle where the hardness gap is large, it suffices for us to make a significantly weaker, *subexponential*, repeated squaring assumption, that is,

$$2^t \text{ repeated squarings (modulo } N = pq) \text{ cannot be done in parallel-time } 2^{t^\varepsilon}$$

More formally:

Assumption 1 (Subexponential Repeated Squaring Assumption)**.** *There exits subexponential functions T, B and a constant c such that for every function $t(\cdot)$ such that $c \log n < t(n) < B(n)$, the following holds: For every size $B(\cdot)$-attacker A with running-time (i.e,. circuit depth) $T(t(\cdot))$, there exists a negligible function μ such that for every $n \in \mathbb{N}$, the probability that A, given g, N where N is a randomly chosen n-bit RSA-modulus, and g is a randomly chosen (or appropriately fixed) element in Z_N^*, can compute $g^{2^{2^t}} \bmod N$ is bounded by $\mu(n)$.*

We remark that, in our eyes, the subexponential repeated squaring assumption is milder than most "standard" subexponential assumptions used in the cryptographic literature (such as e.g., the subexponential DDH assumption, which is a decisional assumption), and has a stronger "win-win" flavor than most cryptographic assumptions: Repeated squaring is a problem that arises naturally in the design of algorithms (e.g., any improvement on repeated squaring would yield improved efficiency for the verification of RSA-based signatures.)

We finally mention that the time-lock puzzle needed for our construction can also be based the existence of a parallel-time hard language and indistinguishability obfuscation (with subexponential security) by the work of Bitansky *et al.* [BGJ+16].)

Towards Non-interactive Non-malleable Commitments We also address the question of whether fully non-interactive (i.e., single-message) non-malleable commitments are possible. We show that if we replace the assumption of the existence of ZAPs (i.e., two-message witness indistinguishability) with non-interactive witness indistinguishable proofs (NIWI) [BOV05, GOS06, BP15], and the existence of families of collision-resistant hash functions for a *single*, *uniform*, collision-resistant hash function [BP04, Rog06], then a slightly modified *non-interactive* version of our protocol satisfies concurrent non-malleability w.r.t. *uniform attackers*: Basically, the first message of our two-round protocol only contains the first message of the ZAP, and the index of the hash function, so by relying on a NIWI and a single hash function (secure against uniform subexponential-time attackers), the first message can be skipped.

Theorem 2 (Main Theorem, Informal)**.** *Let T and B be two arbitrary subexponential functions. Assume the existence of non-interactive commitments, NIWI, a uniform collision-resistant function, all with subexponential-security, and the existence of a (T, B)-time-lock puzzle. Then, there exists a one-message concurrent non-malleable commitment secure w.r.t. uniform polynomial-time adversaries.*

We leave open the question of whether we can get a non-interactive non-malleable commitment w.r.t. also non-uniform attackers.

A Remark on "Sub-subexponential" Security Let us finally mention that although for the simplicity of notation we rely on subexponential hardness assumption, our actual proof reveals that we only need to rely on "sub sub-exponential" hardness assumption for all the primitives we rely on: namely, we only require security to hold w.r.t. attackers of size (and depth) $2^{n^{1/\log\log n}}$ (and in fact, even slightly less).

Why Time-Lock Puzzles? Our Ideas In a Nut Shell. In cryptography, the power, or *resource*, of attackers is usually measured by their running-time when represented as Turing machines, or equivalently by their circuit-size when represented as circuits. Time-lock puzzles, and more generally time-released cryptography [May93, DN93, JJ99, Nak12, BN00], on the other hand, measure the resource of attackers by their parallel running-time or equivalently by their circuit-depth. Our 2-round non-malleable commitments crucially rely on the synergy of these two types of resources. The key idea is, instead of measuring the hardness of commitment schemes in a single "axis" of resource, measure the hardness in two axes, one refers to circuit-size and the other to circuit-depth. By doing so, we can construct a pair of commitment schemes $\mathsf{Com}_1, \mathsf{Com}_2$ that are simultaneously harder than the other, in different axes. In particular, Com_2 is harder in the axis of *circuit-size*, in the sense that Com_1 admits an extractor of size S while Com_2 is secure against all circuits of size S; on the other hand, Com_1 is harder in the axis of *circuit-depth*, in the sense that it admits an extractor of depth D (and some size S) while Com_1 is hiding against all circuits with depth D (and size S). Such a pair of commitment schemes that are mutually harder than each other already has a weak flavor of non-malleability, which can then be amplified to achieve full-fledged non-malleability. More precisely, we transform the aforementioned commitment schemes, which are non-malleable w.r.t. short "tags" to that for much longer "tags" (explained below), while keeping two rounds. A step in the transformation lifts non-malleability in the stand-alone setting to that in the concurrent setting.

1.2 Concurrent and Independent Work

A concurrent and independent, beautiful, work by Khurana and Sahai (KS) [KS17a, KS17b] also presents a construction of 2-round non-malleable commitments from subexponential "standard-type" assumptions. The results, however, are incomparable, both in terms of assumptions, and also in terms of the acheived results (and use significantly different techniques).

In terms of the acheived results, our protocols satisfies *full* concurrent non-malleability, whereas the KS protocol only satisfies "bounded-concurrent" non-malleability—which is a weaker notion of concurrent non-malleability where the number of sessions is a *a-priori bounded* by some predetermined polynomial in the security parameter; in particular, the communication complexity of their protocol grows super linearly with the bound on the number of sessions, and the complexity assumptions they rely on need to be parametrized by it. Additionally, we also present a fully non-interactive protocol, whereas their technique appears to be inherently limited to two-round protocols.

In terms of assumptions, the key difference is that KS does not rely on time-lock puzzles but rather on the existence of certain 2-round secure two-party computation protocols (with super-polynomial-time simulation security); they also claim that such protocols can be constructed based on the subexponential DDH assumption, or the subexponential QR assumption. These assumptions are incomparable to the subexponential repeated squaring assumption. While DDH and QR are clearly more typical assumptions in the literature on cryptographic protocols, as we mentioned above, the repeated squaring assumption is, in our eyes, perhaps an even more natural computational problem that has been extensively studied over the years. On a qualitative level, it is also a search assumption (and thus our construction of non-malleable commitments can be based on search assumptions), whereas the KS construction (due to the above DDH, or QR, assumption) relies on "decisional assumptions". (Finally, on a quantitative level, as mentioned above, we actually only need to rely on "sub-subexponential" hardness assumptions, whereas KR seem to need subexponential hardness assumptions.)

1.3 Organization

In Section 2, we give a detailed overview of our approach for constructing 2-round non-malleable commitments. In Section 3, we provide preliminaries and definitions. Section 4 presents a family of basic commitment schemes that are mutually harder than each other at different axis, we call them size-robust, depth-robust and size-and-depth robust commitments. Using these basic commitment schemes, in Section 5, we construct a commitment scheme for short identities that satisfy a weaker notion of non-malleability that we formalize as non-malleability w.r.t. extraction. In Section 6, we present a non-malleability strengthening technique that increases the length of identities exponentially, and lifts non-malleability w.r.t. extraction in the stand-alone setting to both non-malleability w.r.t. extraction and standard non-malleability in the concurrent setting. In Section 7, we construct 2-round non-malleable commitment scheme for n-bit identities, by iteratively applying the amplification technique in Section 6 to the basic scheme in Section 5. Finally in Section 8, we show how to remove the first-message in our 2-round non-malleable commitment when the attackers are restricted to be uniform Turing machines.

2 Overview

Every statistically binding commitment scheme is *hiding* against polynomial-sized circuits, while *extractable* by some exponential-sized circuit (such an extractor is guaranteed to exist since one can always find the committed value by brute force). In this work, we pay special attention to the *gap* between the "resources" of attackers and that of extractors. Moreover, we crucially rely on the synergy between different resources — in particular, *circuit-size* and *circuit-depth*, which are captured by the following two basic types of commitment schemes:

Size-Robust Commitments are parametrized versions of classical commitments: An (S, S')-*size-robust commitment* is hiding against any size-$\mathsf{poly}(S)$ attackers, and extractable by some size-S' extractor, for an $S' = S^{\omega(1)}$ denoted as $S' >> S$. Importantly, the extractor has large size, but *shallow* polynomial depth. Such extractors can be implemented using the naïve brute force strategy of enumerating all possible decommitments, which a is time-consuming but highly-parallelizable task.

Depth-Robust Commitments are natural analogues of size-robust commitments, but with respect to the resource of circuit-depth. A (D, D')-*depth-robust commitment* is hiding against any depth-$\mathsf{poly}(D)$ circuits with size up to a large upper bound B, and extractable by some *size-*D' extractor for a $D' >> D$ that necessarily has a depth super-polynomially larger than D. In this work, we consider a subexponential size upper bound $B = 2^{n^\varepsilon}$ for some constant $\varepsilon > 0$; for simplicity of exposition, we ignore this upper bound in the rest of this overview (see Section 4 for more detail).

Size-Robust Commitments from Subexponential Injective OWFs. Size-robust commitments can essentially be instantiated using any off-the-shelf commitment schemes that are subexponential secure, by appropriately scaling the security parameter to control the levels of security and hardness for extraction. Take the standard non-interactive commitment scheme from any injective one-way function f as an example: A commitment to a bit b is of form $f(r), h(r) \oplus b$, consisting of the image $f(r)$ of a random string r of length n, and the committed bit b XORed with the hard-core bit $h(r)$. Assuming that f is subexponentially hard to invert, the commitment is hiding against all size-2^{n^ε} circuits for some constant $\varepsilon > 0$, while extractable in size 2^n (ignoring polynomial factors

in n) and polynomial depth. By setting the security parameter n to $(\log S)^{1/\varepsilon}$, we immediately obtain a (S, S')-size robust commitment for $S' = 2^{\log S^{1/\varepsilon}}$.

Depth-Robust Commitments from Time-Lock Puzzles. Depth-robust commitments are naturally connected with cryptographic objects that consider parallel-time complexity, which corresponds to circuit-depth. When replacing subexponentially-hard one-way functions in the above construction with time-lock puzzles, we immediately obtain depth-robust commitments:

- To commit to a bit b, generate a puzzle puz with a random solution s and a designated level of hardness t, and hide b using the Goldreich-Levin hard-core bit, producing $C = (\mathsf{puz}, r, \langle r, s\rangle \oplus b)$ as the commitment.

- To decommit, the committer can simply reveal the puzzle solution s together with the random coins ρ used for generating the puzzle. The receiver verifies that the puzzle is honestly generated with solution s, and uses s to recover the committed bit b.

Since the time-lock puzzle solution s is hidden against adversaries in parallel-time $T(t)$ (and overall time $B(n)$), the commitments are hiding against depth-$T(t)$ adversaries (with size up to $B(n)$). Moreover, since the puzzles can be "forcefully" solved in time 2^t, the committed values can be extracted in size 2^t. This gives a $(T, 2^t)$-depth-robust commitment.

Next, we show how to compose the basic size-robust and depth-robust commitment schemes to overcome Pass's impossibility result on 2-round non-malleable commitments.

2.1 Towards Overcoming the Impossibility Result

In the literature, there are two formulations of non-malleable commitments, depending on whether the commitment scheme uses players' *identities* or not. The formulation with identities, adopted in this work, assume that the players have identities of certain length ℓ, and that the commitment protocol depends on the identity of the committer, which is also referred to as the *tag* of the interaction. Non-malleability ensures that, as long as the tags of the left and right commitments are different (that is, the man-in-the-middle does not copy the identity of the left committer), no man-in-the-middle attackers can "maul" a commitment it receives *on the left* into a commitment of a related value it gives *on the right*. This is this is formalized by requiring that for any two values v_1, v_2, the values the man-in-the-middle commits to after receiving left commitments to v_1 or v_2 are indistinguishable.

The length ℓ of the tags can be viewed as a quantitative measure of how non-malleable a scheme is: A ℓ-bit tag non-malleable commitment gives a family of 2^ℓ commitment schemes — each with a hardwired tag — that are "mutually non-malleable" to each other. Therefore, the shorter the tags are, the easier it is to construct such a family. Full-fledged non-malleable commitments have tags of length equal to the security parameter $\ell = n$, and hence corresponds to a exponentially sized family. However, when the number of communication rounds is restricted to 2, Pass [Pas13] showed that even the weakest non-malleable commitment for just *1-bit tags*, corresponding to a size 2 family, cannot be reduced from falsifiable assumptions, via a polynomial-time black-box reduction.

One-Sided Non-Malleability via Complexity Leveraging. It is well known that *one-sided non-malleability* can be achieved easily via complexity leveraging. One-sided non-malleability only prevents mauling attacks when the tag of the left commitment is "larger than" the tag of the right

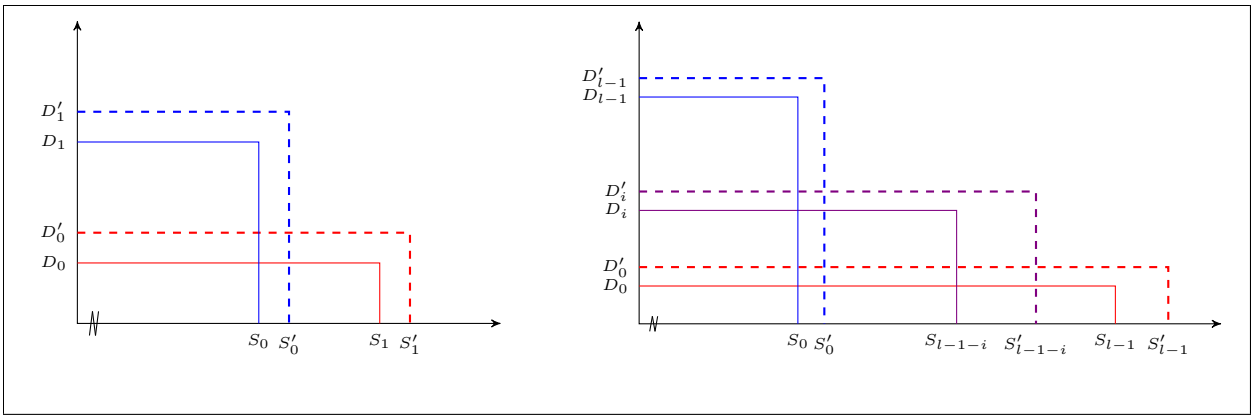


Figure 1: (left) A 1-bit tag based commitment scheme: The tag-0 (resp., tag-1) commitment scheme is hiding for circuits of depth below D_0 (resp., D_1) *OR* size below S_1 (resp., S_0), represented by the solid line joining D_0 (resp., D_1) and S_1 (resp., S_0). The tag-0 (resp., tag-1) commitment scheme admits an extractor of depth at most D'_0 (resp., D'_1) and size at most S'_1 (resp., S'_0). (right) This is a generalization of the 1-bit tag commitment scheme to $\log l$-bits tags, where for tag-i the commitment scheme is hiding for circuits of depth below D_i *OR* size below S_{l-1-i} and exhibits an extractor of depth at most D'_i and size at most S'_{l-1-i}.

commitment [1]. In the simple case of 1-bit tags, this requires the commitment for tag 1 (on the left) to be non-malleable w.r.t. the commitment for tag 0 (on the right), which holds if the tag-1 commitment is "harder" than the tag-0 commitment. For example, if the tag-1 commitment is (S_1, S'_1)-size-robust while the tag-0 commitment is (S_0, S'_0)-size-robust for some $S_0 << S'_0 << S_1 << S'_1$, then one can extract the right committed value using a size-S_1 extractor, while the left committed value still remain hidden. Therefore, the right committed value must be (computationally) independent of the left. Similarly, we can also achieve one-sided non-malleability using depth-robust commitments, by using a (D_1, D'_1)-depth robust commitment scheme for tag 1 and a (D_0, D'_0)-depth robust commitment scheme for tag 0, for some $D_0 << D'_0 << D_1 << D'_1$.

However, simple complexity leveraging is inherently limited to one-sided non-malleability, since when only one resource is considered, the tag-1 commitment cannot be both harder and easier than the tag-0 commitment.

Two Resources for (Two-Sided) Non-Malleability. Therefore, our key idea is using two resources to create two "axis", such that, the tag-1 commitment and tag-0 commitment are simultaneously "harder" than the other, but, with respect to different resources. This is achieved by combining the basic size-robust and depth-robust commitment schemes in the following simple way.

Basic 1-bit Tag Non-Malleable Commitment:
For some $D_0 << D'_0 << D_1 << D'_1 << S_0 << S'_0 << S_1 << S'_1$,

- a tag-0 commitment to a value v consists of commitments to two random secret shares α, β of v, such that, $v = \alpha + \beta$, where the first share is committed under a (D_0, D'_0)-depth-robust commitment scheme and the second under a (S_1, S'_1)-size-robust commitment scheme, and

[1] The choice that the left tag is smaller than the right tag is not important. One could also require the opposite that the left tag is larger than the right tag. The limitation is that the design of the commitments depends on this arbitrary decision.

- a tag-1 commitment to v, on the other hand, uses a (D_1, D_1')-depth-robust commitment scheme to commit to the first share and a (S_0, S_0')-size-robust commitment scheme to commit to the second share.

Thus, the tag-1 commitment is harder w.r.t. circuit-depth, while the tag-0 commitment is harder w.r.t. circuit-size. Leveraging this difference, one can extract from a tag-0 commitment (on the right) without violating the hiding property of a tag-1 commitment (on the left), and vice versa — leading to two-sided non-malleability. More specifically, the committed values in a tag-0 commitment can be extracted in depth D_0' and size S_1' by extracting both secret shares from the size- and depth-robust commitments contained in it. Yet, adversaries with such depth and size cannot break the (D_1, D_1')-depth-robust commitment contained in a tag-1 commitment; thus, the value committed to in the tag-1 commitment remains hidden. On the flip side, the committed value in a tag-1 commitment can be extracted in depth D_1' and size S_0', and, similarly, adversaries with such depth and size do not violate the hiding of a tag-0 commitment, due to the fact that the size-robust commitment contained in it is hiding against size-S_1 adversaries.

In summary, combining the two types of commitment schemes gives us depth-and-size robust commitment schemes: A $(D \vee S, D' \wedge S')$-robust commitment is hiding against circuits with depth below D *or* size below S, while extractable by some circuit with depth D *and* size S, as illustrated in Figure 1 (left). In this language, a tag-0 commitment is $(D_0 \vee S_1, D_0' \wedge S_1')$-robust while a tag-1 commitment is $(D_1 \vee S_0, D_1' \wedge S_0')$-robust. They are mutually non-malleable, because the extractor for one falls into the class of adversaries that the other is hiding against.

The Subtle Issue of Over-Extraction The above argument captures our key idea, but is overly-simplified. It implicitly assumes that the size- and depth-robust commitments are extractable in the perfect manner: 1) Whenever a commitment is valid, in the sense that there exists an accepting decommitment, the extractor outputs exactly the committed value, otherwise, 2) when the commitment is invalid, it outputs $\perp$. Such strong extractability ensures that to show non-malleability that the right *committed* value is independent of the left committed value, it suffices to show that the right *extracted* value is independent of the left committed value, as argued above.

However, our depth-robust commitments from time-lock puzzles do not satisfy such strong extractability. [2] In particular, they do not satisfy the second property above: When commitments are invalid, the extractor can output arbitrary values — this is known as "over-extraction". Over-extraction traces back to the fact that only *honestly generated* time-lock puzzles (*i.e.*, in the domain of the puzzle generation algorithm) are guaranteed to be solvable in certain time. There is no guarantee for ill-generated puzzles, and no efficient procedure for deciding whether a puzzle is honestly generated or not. Observe that this is the case for the time-lock puzzles proposed by Rivest, Shamir, and Wagner [RSW96], since given a puzzle $(s + a^{2^{2^t}} \bmod N,\ N)$ one can extract s using 2^t squaring modular N, but cannot obtain a proof that N is a valid RSA-modulus; this is also the case for the other puzzle construction [BGJ+16]. As a result, the extractor of our depth-robust commitments that extracts committed values via solving time-lock puzzles, provides no guarantees when commitments are invalid.

This means that our basic 1-bit tag commitment scheme is over-extractable, and the argument above that reasons about the right extracted value fails to establish non-malleability. Nevertheless, the basic scheme does satisfy a variant of non-malleability that we call *non-malleability w.r.t. extraction*, which ensures that the value *extracted* from the right commitment is independent of the left committed value. When a commitment scheme is perfectly-extractable, this new notion is

[2] Our size-robust commitments from injective one-way functions do satisfy such strong extractability.

equivalent to standard non-malleability (w.r.t. commitment), but with over-extraction, it becomes incomparable. The issue of over-extraction has appeared in the literature (*e.g.*, [Wee10, Kiy14]), standard methods for eliminating it requires the committer to additionally prove the validity of the commitment it sends, using for instance zero-knowledge protocols or cut-and-choose techniques. However, these methods take more than 2 rounds of interaction, and does not apply here.

2.2 Full-Fledged Non-Malleable Commitments

At this point, we face two challenges towards constructing full-fledged non-malleable commitments:

- *Challenge 1:* We need to go from non-malleability w.r.t. extraction to non-malleability w.r.t. commitment in 2 rounds. Resolving this challenge would give a 2-round 1-bit tag non-malleable commitment scheme.

- *Challenge 2:* The next challenge is going beyond two tags, towards supporting an exponential 2^n number of tags.

  It is easy to generalize our basic 1-bit tag commitment scheme to handle arbitrary l tags, if there exists a "ladder" of l commitment schemes with increasing levels of depth-robustness, and another "ladder" of l schemes with increasing levels of size-robustness. Concretely, the i'th schemes are respectively (D_i, D'_i)-depth robust and (S_i, S'_i)-size robust, for some

  $$\cdots << D_i << D'_i << \cdots << D_l << D'_l \;\; << \;\; S_0 << S'_0 \cdots << S_i << S'_i << \cdots \; .$$

  A commitment with tag $i \in \{0, \cdots, l-1\}$ combines the i'th (D_i, D'_i)-depth-robust scheme and the $(l-1-i)$'th (S_{l-1-i}, S'_{l-i-1})-size-robust scheme to commit to a pair of secret shares of the committed value. This gives a family of l mutually non-malleable commitment schemes, as illustrated in Figure 1 (right).

  To directly obtain full-fledged non-malleable commitments, we need an exponential number of levels $l = 2^n$ of depth- and size-robustness, which is, however, impossible from the underlying assumptions. From subexponentially hard injective one-way functions, we can instantiate at most $O(\log n / \log\log n)$ levels of size-robustness, and similarly, from subexponentially parallel-time hard time-lock puzzles, we can instantiate $O(\log n / \log\log n)$ levels of depth-robustness. Therefore, we need to amplify the number of tags.

We address both challenges using the a single transformation.

2-Round Tag Amplification Technique: We present a transformation that converts a 2-round l-tag commitment scheme that is non-malleable w.r.t. extraction, into a 2-round 2^{l-1}-tag commitment scheme that is both non-malleable w.r.t. extraction and w.r.t. commitment. The output protocol can be further transformed to achieve concurrent non-malleability.

With the above transformation, we can now construct full-fledged non-malleable commitment. Start from our basic scheme for a constant $l_0 = O(1)$ number of tags that is non-malleable w.r.t. extraction; apply the tag-amplification technique *iteratively for* $m = O(\log^* n)$ *times* to obtain a scheme for $l_m = 2^n$ tags that is both non-malleable w.r.t. extraction and w.r.t. commitment.

Previously, similar tag-amplification techniques were presented by Lin and Pass [LP09] and Wee [Wee10]. Our transformation follows the same blueprint, but differ at two important aspects. First, our transformation starts with and preserves non-malleable w.r.t. extractability, which is not considered in their work. Second, their amplification techniques incur a constant additive overhead

in the round complexity of the protocol, whereas our transformation keeps the number of rounds invariant at 2. To do so, our amplification step combines ideas from previous works with the new idea of using our depth-and-size robust commitments to create different 2-round sub-protocols that are mutually "non-malleable" when executed in parallel, in the sense that the security of one sub-protocol remains intact even when the security of another is violated by force.

Our 2-Round Tag-Amplification Technique in More Detail. Similar to [LP09, Wee10], the transformation proceeds in two steps:

- First, amplify the security of a scheme from (*one-one*) non-malleability w.r.t. extraction to *one-many* non-malleability w.r.t. extraction and commitment, which, following a proof in [LPV08], implies *concurrent* (or many-many) non-malleability w.r.t. extraction and commitment. (This is why our final protocol can be made concurrently non-malleable.) Here, one-many and concurrent non-malleability w.r.t. extraction or commitment naturally generalize standard non-malleability to the setting where the man-in-the-middle concurrently receives one or many commitments on the left and gives many commitments on the right, and ensures that the joint distribution of the values extracted from or committed in right commitments is independent of the value(s) committed in the left.

- Next, apply the "log-n trick" by Dolev, Dwork and Naor [DN00] to amplify the number of tags supported from l to 2^{l-1} at the price of losing concurrent security, yielding a protocol that is (*one-one*) non-malleable w.r.t. extraction and commitment.

The main technical challenges lie in the first step. We briefly review the LP approach. At a high-level, they construct one-many non-malleable commitment following the Fiat-Shamir paradigm: The receiver starts by setting up a *hidden* "trapdoor" t. The sender commits to a value v using an arbitrary (potentially malleable) 2-message commitment scheme, followed by committing to 0^n using a (one-one) non-malleable commitment and proving using *many* witness-indistinguishable proofs of knowledge (WIPOK) that either it knows a decommitment to v *or* it knows a decommitment of the non-malleable commitment to the trapdoor t; the former, called the honest witness, is used by the honest committer, while the latter, called the fake witness, is used for simulation.

The LP protocol arranges all components — the trapdoor-setup, commitment to v, non-malleable commitment (for trapdoor), and every WIPOK — *sequentially*. To compress the protocol into 2 rounds, we run all components in *parallel*, and replace multiple WIPOK proofs with a single 2-round ZAP proof.

Unfortunately, arranging all components in parallel renders the proof of one-many non-malleability in LP invalid. They designed a sequence of hybrids in which different components in the (single) left interaction are gradually switched from being honestly generated to simulated, while maintaining two invariants regarding the (many) right interactions. First, the *soundness* condition states that the man-in-the-middle never commits to a trapdoor in any right interaction. Second, in every right interaction, there is always a WIPOK that can be rewound to extract the value committed to in this interaction, without rewinding the left component being changed; the value extracted must be a valid decommitment since the fake witness does not exist by the soundness invariant — this establishes *strong extractability*. The second invariant is true because the LP protocol contains sufficiently many sequential WIPOKs so that there is always a proof that does not interleave with the left-component being changed. The first invariant, on the other hand, relies not only on the non-malleability of the input commitment scheme, but also on its "robustness" to other components that have a small fixed k number of interactions (such as 2-message commitment and WIPOK).

The robustness captures "non-malleability" w.r.t. other protocols, and is achieved by embedding more than k rewinding slots in the input commitment scheme.

In our 2-round protocol, we cannot afford to have many rewinding slots for extraction, nor for establishing non-malleability between different components. Naturally, we resort to our size-and-depth robust commitments, which can be made mutually non-malleable w.r.t. extraction by setting the appropriate profiles of size-and-depth robustness. Using a family of 4 such schemes, we mimic the LP proof in the following (overly-simplified) manner: In every hybrid, in the left interaction, either a size-and-depth robust commitment or the non-malleable commitment is changed, while on the right, values are extracted from a *different* size-and-depth robust commitment and from the non-malleable commitment. To show that the left interaction remains indistinguishable despite of extraction, we rely on the mutual non-malleability of the size-and-depth robust schemes, but also seems to need the non-malleable commitment and the size-and-depth robust commitments to be mutually non-malleable, which unfortunately does not hold.

Let us explain. It turns out that our basic non-malleable commitment schemes for short tags, and all intermediate schemes produced by the tag-amplification technique are only secure against circuits with *both* bounded-size *and* bounded-depth. In contrast, the depth-and-size robust commitments are secure against circuits with *either* bounded-size *or* bounded-depth. This qualitative difference in adversarial circuit classes prevents them from being mutually non-malleable. To get around this, we instead rely on a "cycle of non-malleability" that consists of the non-malleable commitment scheme and two depth-and-size robust commitment schemes, satisfying that the first scheme is non-malleable to the second, the second non-malleable to the third, and the third to the first. Such a cycle turns out to be sufficient for our proof to go through.

One final technicality is that in order to create the cycle of non-malleability, the hardness of the two size-and-depth robust commitments must be set appropriately according to that of the non-malleable commitment scheme. Furthermore, the non-malleable commitment scheme produced by the above transformation has weaker security than the input scheme. As a result, to iteratively apply the tag-amplification technique for $O(\log^* n)$ times, we need $O(\log^* n)$ levels of depth- and size-robustness. This can be easily instantiated using subexponentially secure non-interactive commitment schemes and time-lock puzzles as stated in Theorem 1.

See Section 6 for more details on our tag amplification and its security proof.

3 Preliminaries

3.1 Basic Notation

We denote n as the security parameter. For $n \in \mathbb{N}$, by $[n]$ we denote the set $\{0, \ldots, n-1\}$. If v is a binary string then $|v|$ denotes the length of the string and $v[i]$ is the ith bit of v, for $0 \leq i \leq |v|-1$. We use $||$ as the string concatenation operator. For any probability distribution D, $x \leftarrow D$ denotes sampling an element from the distribution D and assigning it to x. However, for a finite set Q, $x \leftarrow Q$ denotes sampling an element from the set Q uniformly and randomly, and assigning it to x. We model algorithms as uniform TMs. We use the abbreviation PPT to denote probabilistic-polynomial time. $\mathcal{P}/\mathrm{poly}$ is the set of all non-uniform polynomial size circuits. We say that a function $\nu : \mathbb{N} \to \mathbb{R}$ is negligible, if for every constant $c > 0$ and for sufficiently large $n \in \mathbb{N}$ we have $\nu(n) < n^{-c}$. For functions d, S defined over $\mathbb{N}$, we say that $d < S$ (resp. $d \leq S$) if for every $n \in \mathbb{N}$, $d(n) < S(n)$ (resp. $d(n) \leq S(n)$). Furthermore, we say that $d << S$ if for every polynomial poly, $\mathsf{poly}(d) < S$.

3.2 Circuit Classes

We define the following circuit classes which are going to be used throughout this work. For the following definitions, consider $n \in \mathbb{N}$ and let d, S and S^* be some non-decreasing functions defined on $\mathbb{N}$ such that $d \leq S << S^*$.

Definition 1 (Depth $\wedge$ size-restricted circuits)**.** *$\mathcal{C}^{\wedge}_{d,S}$ is the set of all non-uniform circuits $C = \{C_n\}_{n\in\mathbb{N}}$ such that there exists a polynomial* poly *such that for all $n \in \mathbb{N}$,*

$$\begin{aligned} & \mathsf{dep}(C_n) < \mathsf{poly}(d(n)) \\ \textit{and} \quad & \mathsf{size}(C_n) < \mathsf{poly}(S(n)) \ , \end{aligned}$$

where $\mathsf{dep}(C_n)$ *and* $\mathsf{size}(C_n)$ *denote the depth and the size of the circuit C_n respectively.*

Throughout this work, we use S^* to denote some pre-defined upper bound on the size of any circuit considered in this work. Furthermore, when we are only concerned with restricting the depth of the circuits, whose size can be as large as the upperbound $\mathsf{poly}(S^*)$ for any polynomial poly, we simply refer to the circuit class $\mathcal{C}^{\wedge}_{d,S^*}$ as $\mathcal{C}_d$.

Definition 2 (Depth-restricted circuits)**.** *$\mathcal{C}_d$ is the set of all non-uniform circuits $C = \{C_n\}_{n\in\mathbb{N}}$ such that there exists a polynomial* poly *such that for all $n \in \mathbb{N}$,*

$$\begin{aligned} & \mathsf{dep}(C_n) < \mathsf{poly}(d(n)) \\ \textit{and} \quad & \mathsf{size}(C_n) < \mathsf{poly}(S^*(n)) \ . \end{aligned}$$

Furthermore, when we want to only restrict the size of the circuits, allowing for the depth to be as large as the size, we refer to the circuit class $\mathcal{C}^{\wedge}_{S,S}$ as $\mathcal{C}_S$.

Definition 3 (Size-restricted circuits)**.** *$\mathcal{C}_S$ is the set of all non-uniform circuits $C = \{C_n\}_{n\in\mathbb{N}}$ such that there exists a polynomial* $\mathsf{poly}(\cdot)$ *such that for all $n \in \mathbb{N}$,*

$$\mathsf{dep}(C_n) \leq \mathsf{size}(C_n) < \mathsf{poly}(S^*(n)) \ .$$

Definition 4 (Depth $\vee$ size-restricted circuits)**.** *$\mathcal{C}^{\vee}_{d,S}$ is the set of all non-uniform circuits $C = \{C_n\}_{n\in\mathbb{N}}$ such that either $C \in \mathcal{C}_d$ or $C \in \mathcal{C}_S$.*

Remark 1. *The classes of circuits $\mathcal{C}$ (namely, $\mathcal{C}_d$, $\mathcal{C}_S$, $\mathcal{C}^{\vee}_{d,S}$ and $\mathcal{C}^{\wedge}_{d,S}$) considered in this work are such that $S \geq d >> n$, that is, all d and S are super-polynomials. For any circuit $C \in \mathcal{C}$, on composing with a circuit $P \in \mathcal{P}/\mathrm{poly}$, it is easy to see that the resulting circuit is also in the class $\mathcal{C}$. Therefore, we say that the circuit class $\mathcal{C}$ is closed under composition with $\mathcal{P}/\mathrm{poly}$. This fact is going to be important in the rest of this work.*

Below, we define standard cryptographic primitives w.r.t. a general circuit class $\mathcal{C}$, requiring that any adversary in $\mathcal{C}$ has negligible advantage in breaking the security of the primitive. When $\mathcal{C} = \mathcal{P}/\mathrm{poly}$, we say that the primitive is computationally secure and when $\mathcal{C}$ is the set of non-uniform circuits whose size is bounded by $2^{n^{\varepsilon}}$ for some constant $\varepsilon < 1$, we say that the primitive is subexponentially secure.

3.3 Indistinguishability and One-wayness

Definition 5 ($\mathcal{C}$-indistinguishability)**.** *Two ensembles $\{A_{n,y}\}_{n\in\mathbb{N},y\in Y_n}$ and $\{B_{n,y}\}_{n\in\mathbb{N},y\in Y_n}$ are said to be $\mathcal{C}$-indistinguishable, if for every non-uniform circuit $D = \{D_n\}_{n\in\mathbb{N}} \in \mathcal{C}$, there exists a negligible function $\nu(\cdot)$ such that for every $n \in \mathbb{N}$, $y \in Y_n$:*

$$|\mathsf{Pr}\left[a \leftarrow A_{n,y} : D_n(y,a) = 1\right] - \mathsf{Pr}\left[b \leftarrow B_{n,y} : D_n(y,b) = 1\right]| \leq \nu(n) \ .$$

Definition 6 (One-way functions)**.** *A function $f : \{0,1\}^* \to \{0,1\}^*$ is called a $\mathcal{C}$-secure one-way function if the following hold:*

1. *There exists a deterministic polynomial-time algorithm that on input s in the domain of f outputs $f(s)$.*
2. *For every $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}$ there exists a negligible function $\nu(\cdot)$ such that for every $n \in \mathbb{N}$,*

$$\mathsf{Pr}\left[s \leftarrow \{0,1\}^n, s' \leftarrow A_n(f(s)) : f(s') = f(s)\right] \leq \nu(n) \ .$$

In this work, we will use a one-way function that is a permutation which is subexponentially secure.

3.4 Witness Relation, ZAP and NIWI

Definition 7 (Witness Relation)**.** *A witness relation or relation (for short) for a language $L \in \mathcal{NP}$ is a binary relation $\mathcal{R}_L$ that is polynomially bounded, polynomial time recognizable and characterizes L by $L = \{x : \exists w \ s.t. \ (x,w) \in \mathcal{R}_L\}$.*

We say that w is a witness for the membership of $x \in L$ if $(x,w) \in \mathcal{R}_L$. We will also let $\mathcal{R}_L(x)$ denote the set of witnesses for the membership of $x \in L$; that is, $\mathcal{R}_L(x) = \{w : (x,w) \in \mathcal{R}_L\}$.

ZAPs are two-message public coin witness indistinguishable proofs defined as follows.

Definition 8 (ZAP [DN00])**.** *A pair of algorithms $(\mathcal{P},\mathcal{V})$, where $\mathcal{P}$ is PPT and $\mathcal{V}$ is (deterministic) polytime, is a $\mathcal{C}$-ZAP for an $\mathcal{NP}$ relation $\mathcal{R}_L$ if it satisfies:*

1. *Completeness: There exists a polynomial $l(\cdot)$ such that for every $(x,w) \in \mathcal{R}_L$,*

$$\mathsf{Pr}\left[r \leftarrow \{0,1\}^{l(|x|)}, \pi \leftarrow \mathcal{P}(x,w,r) : \mathcal{V}(x,\pi,r) = 1\right] = 1 \ .$$

2. *Adaptive soundness: There exists a negligible function $\nu(\cdot)$ such that for every malicious (potentially unbounded) prover $\mathcal{P}^*$ and every $n \in \mathbb{N}$,*

$$\mathsf{Pr}\left[r \leftarrow \{0,1\}^{l(n)}, (x,\pi) \leftarrow \mathcal{P}^*(r) : x \in \{0,1\}^n \setminus L_n \wedge \mathcal{V}(x,\pi,r) = 1\right] \leq \nu(n) \ .$$

3. *$\mathcal{C}$-witness indistinguishability: For any sequence $\{(x_n, w_n^1, w_n^2, r_n)\}_{n\in\mathbb{N}}$ such that for every $n \in \mathbb{N}$, $x_n \in L_n$, $w_n^1, w_n^2 \in \mathcal{R}_L(x_n)$ and $r_n \in \{0,1\}^{l(n)}$, the following ensembles are $\mathcal{C}$-indistinguishable:*

$$\{\pi_1 \leftarrow \mathcal{P}(x_n, w_n^1, r_n) : (x_n, w_n^1, w_n^2, \pi_1, r_n)\}_{n\in\mathbb{N}} \ ,$$
$$\{\pi_2 \leftarrow \mathcal{P}(x_n, w_n^2, r_n) : (x_n, w_n^1, w_n^2, \pi_2, r_n)\}_{n\in\mathbb{N}} \ .$$

Throughout this work, we will refer to the first message r of ZAP as a_{ZAP} and the second message together with the statement (π, x) as b_{ZAP}.

Dwork and Naor [DN00] were the first to construct a ZAP from trapdoor permutations. They also showed that ZAP for $L \in \mathcal{NP}$ can be based on the weaker assumption of the existence of NIZKs for L.

Theorem 3. *If there exists a $\mathcal{C}$-secure family of trapdoor permutations then there exists a $\mathcal{C}$-ZAP.*

Furthermore, Bitansky and Paneth [BP15] construct ZAP based on the existence of indistinguishability obfuscation (iO) for a certain family of polysize circuits and one-way functions.

NIWIs are non-interactive witness-indistinguishable proofs.

Definition 9 (NIWI [BOV05])**.** *A pair of algorithms $(\mathcal{P}, \mathcal{V})$ where $\mathcal{P}$ is PPT and $\mathcal{V}$ is (deterministic) polytime, is a $\mathcal{C}$-NIWI for an $\mathcal{NP}$ relation $\mathcal{R}_L$ if it satisfies:*

1. *Completeness: For every $(x, w) \in \mathcal{R}_L$,*
$$\mathsf{Pr}\left[\pi \leftarrow \mathcal{P}(x, w) : \mathcal{V}(x, \pi) = 1\right] = 1 \ .$$
2. *Soundness: For every $x \notin L$ and $\pi \in \{0,1\}^{\mathsf{poly}(n)}$:*
$$\mathsf{Pr}\left[\mathcal{V}(x, \pi) = 1\right] = 0 \ .$$
3. *$\mathcal{C}$-witness indistinguishability: For any sequence $\{(x_n, w_n^1, w_n^2)\}_{n\in\mathbb{N}}$ such that for every $n \in \mathbb{N}$, $x_n \in L_n$, $w_n^1, w_n^2 \in \mathcal{R}_L(x_n)$, the following ensembles are $\mathcal{C}$-indistinguishable:*
$$\{\pi_1 \leftarrow \mathcal{P}(x_n, w_n^1) : (x_n, w_n^1, w_n^2, \pi_1)\}_{n\in\mathbb{N}} \ ,$$
$$\{\pi_2 \leftarrow \mathcal{P}(x_n, w_n^2) : (x_n, w_n^1, w_n^2, \pi_2)\}_{n\in\mathbb{N}} \ .$$

Dwork and Naor [DN00] showed the existence of a non-uniform non-constructive NIWI which can be based on their ZAP construction by fixing the first message non-uniformly. Building on their work, Barak, Ong and Vadhan [BOV05] de-randomize the ZAP verifier in [DN00] to give the first NIWI construction. They base their de-randomization technique on the existence of a function in $Dtime(2^{O(n)})$ with non-deterministic circuit complexity $2^{\Omega(n)}$. The ZAP construction from [BP15] can also be de-randomized under the same assumption. Furthermore, Groth, Ostrovsky and Sahai [GOS06] construct a NIWI based on the decisional linear assumption for bilinear groups.

Theorem 4. *We base the existence of NIWI on either of the following assumptions:*

1. *If decisional linear assumption holds for the elliptic curve based bilinear groups in [BF01] against all circuits in class $\mathcal{C}$ then there exists a $\mathcal{C}$-NIWI.*
2. *If $\mathcal{C}$-secure trapdoor permutations exist and there exists a function in $Dtime(2^{O(n)})$ with non-deterministic circuit complexity $2^{\Omega(n)}$ then there exists a $\mathcal{C}$-NIWI.*

3.5 Commitment Schemes

Definition 10 (Commitment scheme)**.** *A commitment scheme $\langle C, R\rangle$ consists of a pair of interactive* PPT *TMs C and R with the following properties:*

1. *The commitment scheme has two stages: a commit stage and a reveal stage. In both stages, C and R receive a security parameter 1^n as common input. C additionally receives a private input $v \in \{0,1\}^n$ that is the string to be committed.*

2. *The commit stage results in a joint output c, called the commitment, a private output for C, d, called the decommitment string. Without loss of generality, c can be the full transcript of the interaction between C and R. Let $n_c = n_c(n)$ denote the maximal length of the commitment c for security parameter n.*

3. *In the reveal stage, committer C sends the pair (v, d) to the receiver R, and R decides to accept or reject the decommitment (v, d) deterministically according to an efficiently computable function* Open*; that is, R accepts iff* $\mathsf{Open}(c, v, d) = 1$.

4. *If C and R do not deviate from the protocol, then R should accept with probability 1 in the reveal stage.*

Furthermore, we say that a commitment c is valid, if there exists a string v and a decommitment string d such that $\mathsf{Open}(c, v, d) = 1$.

Next we define the binding and hiding property of a commitment scheme.

Definition 11 (Statistical binding)**.** *A commitment scheme $\langle C, R\rangle$ is statistically binding if for any committer C^* possibly unbounded, there exists a negligible function $\nu(\cdot)$ such that C^* succeeds in the following game with probability at most $\nu(n)$:*

On security parameter 1^n, C^* first interacts with R in the commit stage to produce a commitment c. Then C^* outputs two decommitments (v_0, d_0) and (v_1, d_1), and succeeds if $v_0, v_1 \in \{0,1\}^n, v_0 \neq v_1$ and R accepts both as decommitments of c.

Furthermore, a commitment scheme is perfectly binding if the probability that C^* succeeds in the above game is 0.

We define the value of any commitment through a function val, that takes as input an arbitrary commitment c and outputs v if c is valid and there exists exactly one value v such that $\mathsf{Open}(c, v, \cdot) = 1$, otherwise it outputs a $\perp$. Note that such a function val may not be efficiently computable.

Definition 12 ($\mathcal{C}$-hiding)**.** *A commitment scheme $\langle C, R\rangle$ is $\mathcal{C}$-hiding if for every non-uniform circuit $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}$, there exists a negligible function $\nu(\cdot)$ such that A succeeds in the following game with probability at most $\nu(n)$ away from $\frac{1}{2}$:*

For security parameter 1^n, A_n outputs a pair of values $v_0, v_1 \in \{0,1\}^n$. C on input v_b, where b is a randomly chosen bit, interacts with A_n to produce a commitment of v_b. A_n outputs a bit b' and wins the game if $b' = b$.

Additionally, we consider commitment schemes that are "tag-based".

Definition 13 (Tag-based commitment scheme)**.** *A commitment scheme $\langle C, R\rangle$ is a tag-based scheme with $t(n)$-bit identities if, in addition to the security parameter 1^n, the committer and receiver also receive a "tag" – a.k.a. identity–*id *of length $t(n)$ as common input.*

When the length $t(n)$ of identities is n, we refer to $\langle C, R\rangle$ as a tag-based commitment scheme.

Definition 14 (Over-extractable commitment scheme)**.** *A statistically binding commitment scheme $\langle C, R\rangle$ is over-extractable w.r.t. extractor $o\mathcal{E} = \{o\mathcal{E}_n\}_{n\in\mathbb{N}}$ if there exists a negligible function $\nu(\cdot)$ such that $\forall n \in \mathbb{N}$, $\forall c \in \{0,1\}^{n_c}$,*

$$\mathsf{Pr}\left[v' \leftarrow o\mathcal{E}_n(c) : c \text{ is valid } \wedge \mathsf{val}(c) \neq v'\right] \leq \nu(n) \ ,$$

where n_c is the maximal length of the commitment generated by $\langle C, R\rangle$ with security parameter n. Furthermore, we say $\langle C, R\rangle$ is (d, S)-over-extractable if the extractor $o\mathcal{E}$ belongs to the circuit class $\mathcal{C}^{\wedge}_{d,S}$.

Remark 2. *Note that the extractor $o\mathcal{E}$ must successfully extract the correct value for any valid commitment (i.e., for which there exists a decommitment), even if the valid commitment is generated by a malicious committer.*

In the rest of the paper whenever we say a commitment scheme, we mean a statistically (perfectly) binding commitment scheme.

THE MAN-IN-THE-MIDDLE (MIM) EXECUTION: Let $\langle C, R\rangle$ be a tag-based commitment scheme. Consider a non-uniform circuit family $A = \{A_n\}_{n\in\mathbb{N}}$. For security parameter n, A_n participates in m-left and m-right interactions[3]. In the left interactions, A_n interacts with C and receives commitments to values $v_1, \ldots, v_m \in \{0,1\}^n$, using identities $\mathsf{id}_1, \mathsf{id}_2, \ldots, \mathsf{id}_m$ of its choice. In the right interactions A_n interacts with R attempting to commit to related values $\tilde{v}_1, \ldots, \tilde{v}_m$, using identities $\tilde{\mathsf{id}}_1, \tilde{\mathsf{id}}_2, \ldots, \tilde{\mathsf{id}}_m$ of its choice. We define the values $\tilde{v}_i$ committed on the right as $\tilde{v}_i = \mathsf{val}(\tilde{c}_i)$ where $\tilde{c}_i$ is the commitment in the ith right interaction. Recall that $\mathsf{val}(c) = \perp$, if c is not valid or that it can be opened to more than one value. Otherwise, $\mathsf{val}(c)$ equals the unique value v it can be opened to. Furthermore, if for any right interaction i, $\tilde{\mathsf{id}}_i = \mathsf{id}_j$ for some j, we set $\tilde{v}_i = \perp$.

We define two different flavours of non-malleability. First we recall the standard notion of non-malleability – a.k.a non-malleability w.r.t. commitment, for (tag-based) commitment schemes. Then, we introduce a new notion called non-malleability w.r.t. extraction for over-extractable commitment schemes.

Non-malleability w.r.t. commitment: Consider a MIM execution with A. Let $\mathsf{mim}^A_{\langle C,R\rangle}(v_1, \ldots, v_m)$ denote the random variable that describes the values $\tilde{v}_1, \ldots, \tilde{v}_m$ that A commits to on the right and the view of A in $\mathsf{MIM}^A_{\langle C,R\rangle}(v_1, \ldots, v_m)$.

Definition 15 (Non-malleability)**.** *A tag-based commitment scheme $\langle C, R\rangle$ is said to be concurrent $\mathcal{C}$-non-malleable if for every circuit family $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}$ participating in $m = \mathsf{poly}(n)$ concurrent interactions, the following ensembles are computationally indistinguishable:*

$$\left\{\mathsf{mim}^A_{\langle C,R\rangle}(v_1^{(1)}, \ldots, v_m^{(1)})\right\}_{n\in\mathbb{N}, v_1^{(1)},\ldots,v_m^{(1)}\in\{0,1\}^n, v_1^{(2)},\ldots,v_m^{(2)}\in\{0,1\}^n} \ ,$$

$$\left\{\mathsf{mim}^A_{\langle C,R\rangle}(v_1^{(2)}, \ldots, v_m^{(2)})\right\}_{n\in\mathbb{N}, v_1^{(1)},\ldots,v_m^{(1)}\in\{0,1\}^n, v_1^{(2)},\ldots,v_m^{(2)}\in\{0,1\}^n} \ .$$

[3] In standard definitions of non-malleability [DDN00, LPV08], the man-in-the-middle adversary is also given some auxiliary information z. In this work, we consider non-malleability against non-uniform circuits, which can be thought of as having z hard-wired in them. This is why we ignore z in our definitions.

Non-malleability w.r.t. extraction: Let $\langle C, R\rangle$ be a tag-based commitment scheme which is over-extractable w.r.t. extractor $o\mathcal{E}$. We say that $\langle C, R\rangle$ is non-malleable w.r.t. extraction if the distributions of the random variable emim defined below are indistinguishable in any two MIM executions with different values committed on the left. Recall that mim describes the view of A and the values $\tilde{v_i}$ that A commits to on the right. However, the random variable $\mathsf{emim}^A_{\langle C,R\rangle}(v_1, \ldots, v_m)$, instead, describes the view of A and the values $\tilde{v_i}'$ which are obtained by running the extractor $o\mathcal{E}$ on input $\tilde{c_i}$ (the ith right commitment); that is, $\tilde{v_i}' \leftarrow o\mathcal{E}_n(\tilde{c_i})$. Note that, if for any right interaction i, $\tilde{\mathsf{id}_i} = \mathsf{id}_j$, for some j, then we set $\tilde{v_i}' = \perp$.

Definition 16 (Non-malleability w.r.t. extraction)**.** *A tag-based commitment scheme $\langle C, R\rangle$ is said to be concurrent $\mathcal{C}$-non-malleable w.r.t. extraction by $o\mathcal{E}$ if the following hold:*

1. *$\langle C, R\rangle$ is over-extractable by $o\mathcal{E}$.*
2. *For every circuit $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}$ participating in $m = \mathsf{poly}(n)$ concurrent interactions, the following ensembles are computationally indistinguishable:*

$$\left\{\mathsf{emim}^A_{\langle C,R\rangle}(v_1^{(1)}, \ldots, v_m^{(1)})\right\}_{n\in\mathbb{N}, v_1^{(1)},\ldots,v_m^{(1)}\in\{0,1\}^n, v_1^{(2)},\ldots,v_m^{(2)}\in\{0,1\}^n},$$
$$\left\{\mathsf{emim}^A_{\langle C,R\rangle}(v_1^{(2)}, \ldots, v_m^{(2)})\right\}_{n\in\mathbb{N}, v_1^{(1)},\ldots,v_m^{(1)}\in\{0,1\}^n, v_1^{(2)},\ldots,v_m^{(2)}\in\{0,1\}^n}.$$

At first glance, it may seem that the new notion — non-malleability w.r.t. extraction, is no more interesting than the standard notion of non-malleability (w.r.t. commitment). After all, an extractor that agrees with the function val establishes that the two notions are equivalent. Most constructions of non-malleable commitment schemes in the literature, in fact, establish non-malleability by building such an extractor in their security proofs. In this work, however, we consider extractors that may not always agree with val and have some *over-extraction*.

Over-extractability guarantees that for valid commitments, the extractor extracts out the committed value with overwhelming probability. However, given an invalid commitment, the value extracted by the extractor can be arbitrary. This inept behaviour of the extractor, on invalid commitments, is what makes the two notions incomparable (in general). For instance, there might exist an adversary A, depending on the value committed on the left, may choose to send invalid transcripts on the right with different probabilities. Such an A certainly breaks the non-malleability of the scheme (w.r.t commitment) but depending on the extractor, A may not violate non-malleability w.r.t. extraction because the extracted values may still be indistinguishable. Furthermore, there might exist an adversary that irrespective of the value on the left always sends invalid commitments on the right. Such an A does not break the non-malleability w.r.t. commitment. But A may violate non-malleability w.r.t. extraction by establishing a co-relation between the value committed on the left and the value that will be over-extracted by the extractor on the right. Hence, the two notions are incomparable. However, if one sets up the decommitment condition (which defines the random variable mim) appropriately then we show that it is possible to base non-malleability w.r.t. commitment on non-malleability w.r.t. extraction. We believe this reduction as one of the main contributions of this work.

We also consider relaxed versions of both non-malleability and non-malleability w.r.t. extraction: one-one, one-many and many-one secure commitment schemes. In one-one (a.k.a. standalone), we consider an adversary A that participates in one left and one right interaction; in one-many A participates in one left and many right; and in many-one, A participates in many left and one right interaction.

3.6 Time-Lock Puzzles

Definition 17 (Time-lock puzzles [BGJ+16])**.** *A (T, B)-time-lock (TL) puzzle is a tuple* $(\mathsf{Gen}, \mathsf{Sol})$ *satisfying the following requirements:*

1. *Syntax:*
   - $Z \leftarrow \mathsf{Gen}(1^n, 1^t, s)$ *is a probabilistic algorithm that takes as input a security parameter n, a solution $s \in \{0,1\}^n$ and a difficulty parameter t and outputs a puzzle Z.*
   - $s \leftarrow \mathsf{Sol}(Z)$ *is a deterministic algorithm that takes as input a puzzle Z and outputs a solution s.*
2. *Completeness: For every security parameter n, difficulty parameter t, solution $s \in \{0,1\}^n$ and puzzle Z in the support of* $\mathsf{Gen}(1^n, 1^t, s)$, $\mathsf{Sol}(Z)$ *outputs s.*
3. *Efficiency:*
   - $Z \leftarrow \mathsf{Gen}(1^n, 1^t, s)$ *is a poly-time algorithm, that is, it runs in time* $\mathsf{poly}(t, n)$.
   - $s \leftarrow \mathsf{Sol}(Z)$ *runs in time* $\mathsf{poly}(2^t)$ *for Z in the support of* $\mathsf{Gen}(1^n, 1^t, \cdot)$.
4. *(T, B)-hardness:* $(\mathsf{Gen}, \mathsf{Sol})$ *is a (T, B)-hard TL puzzle if there exists a constant c such that for every $c \log n < t(n) < B(n)$ and every adversary $A = \{A_n\}_{n \in \mathbb{N}}$ where,*

$$\mathsf{dep}(A_n) \leq T(t) \ ; \ \mathsf{size}(A_n) \leq B(n) \ ,$$

there exists a negligible function ν, such that for every $n \in \mathbb{N}$,

$$\Pr\left[s \leftarrow \{0,1\}^n; \ Z \leftarrow \mathsf{Gen}(1^n, 1^{t(n)}, s); \ s' \leftarrow A_n(Z) : s' = s\right] \leq \nu(n) \ .$$

The first candidate construction of TL puzzles was proposed by Rivest, Shamir and Wagner [RSW96] and is based on the "inherently sequential" nature of exponentiation modulo an RSA integer. Twenty years after their proposal, there still does not exist a (parallelizable) strategy that can solve the puzzle (of difficulty parameter t) in parallel-time $T(t)$ which is significantly less than 2^t. Apart from the variants of RSW puzzles [BN00, GMPY11], the only other construction of TL puzzles was given by Bitansky et al. [BGJ+16] based on succinct randomized encodings for Turing machines (which in turn can be built from indistinguishability obfuscation and one-way functions) and the existence of non-parallelizing languages. These previous works have considered puzzles with strong parameters, that is, puzzles that are parallel-time hard for exponential $T = 2^{\delta t}$ ([BGJ+16]) and even strongly exponential $T = \delta 2^t$ ([BN00, GMPY11]).

However, for our task of constructing 2-round non-malleable commitments, much weaker TL puzzles are sufficient, that is, puzzles that remain hard for only subexponential $T = 2^{t^\delta}$ parallel-time. More precisely, we need a $(T(t) = 2^{t^\delta}, B(n) = 2^{n^\varepsilon})$-TL puzzle for some $0 < \varepsilon, \delta < 1$. We here recall the RSW TL puzzles $\mathsf{RSW} = (\mathsf{Gen}, \mathsf{Sol})$ as a candidate.

- Algorithm $\mathsf{Gen}(1^n, 1^t, s)$:
  1. Select an n-bit RSA modulous $N = pq$.
  2. Compute the mask $y = g^{2^{2^t}} \mod N$ for some element $g \in \mathbb{Z}_N^*$. Note that since the factorization of N is known, Gen can first compute the exponent $e = 2^{2^t} \mod \phi(N)$ and then efficiently compute the mask $y = g^e \mod N$.

3. Mask the solution s with y, that is, $z = (s + y) \mod N$.
4. Return the tuple $Z = (z, N)$ as the puzzle.

- Solver $\mathsf{Sol}(Z = (z, N))$:

1. By 2^t repeated squarings, compute $y = g^{2^{2^t}} \mod N$.
2. Output $(z - y) \mod N$ as the solution.

As discussed in [RSW96], the element g above can either be a fixed element such as 2, or sampled at random.

Next, we discuss that the $\mathsf{RSW} = (\mathsf{Gen}, \mathsf{Sol})$ is a TL puzzle in the sense of Definition 17. It is easy to see that for security parameter n and difficulty parameter t, Gen runs in time $\mathsf{poly}(t, n)$ and Sol runs in time $\mathsf{poly}(2^t)$. Futhermore, we base the (T, B)-hardness of the RSW puzzle on the subexponential RSW assumption as stated in Assumption 1. Informally, it says that for some subexponential functions T and B, and any function t such that $c \log n \leq t(n) \leq B(n)$, $B(n)$-sized adversaries with depth $T(t)$ cannot compute $g^{2^{2^t}} \mod N$. From the discussion presented in Section 1 it follows that if the subexponential RSW assumption holds, then the RSW puzzle as defined above is a (T, B)-hard TL puzzle for some subexponential functions T and B.

Lemma 1. *If the subexponential RSW assumption holds, then there exists subexponential functions T and B, such that, $\mathsf{RSW} = (\mathsf{Gen}, \mathsf{Sol})$ is a (T, B)-hard TL puzzle.*

3.7 Collision-resistant Hash Functions

Definition 18. *A family of non-uniform collision-resistant hash functions (CRH) $\{D_\lambda\}_{\lambda \in \mathbb{N}}$ is a family of distributions such that for every $H \in D_\lambda$, $H : \{0,1\}^{n(\lambda)} \to \{0,1\}^\lambda$ such that $\lambda < n(\lambda)$ satisfying,*

1. *Efficient Computation: There exists a poly-time TM M such that for every $\lambda \in \mathbb{N}$, $H \in D_\lambda$ and $x \in \{0,1\}^{n(\lambda)}$, $M(H, x) = H(x)$.*
2. *$S(\lambda)$-Collision-resistance: For every S-sized family of circuits $\{A_\lambda\}_{\lambda \in \mathbb{N}}$ there exists a negligible function ν such that for every $\lambda \in \mathbb{N}$,*

$$\mathsf{Pr}\left[H \leftarrow D_\lambda, (x_1, x_2) \leftarrow A(H) : x_1 \neq x_2 \wedge H(x_1) = H(x_2)\right] \leq \nu(\lambda) \tag{1}$$

Moreover, a family of uniform *collision resistant hash function (CRH) is as defined above, except that i) the distribution D_λ always outputs a single function H_λ, and ii) $S(\lambda)$-collision resistence only holds against attackers that are $S(\lambda)$-time* uniform *Turing machines. We denote such a family as $\{H_\lambda\}_{\lambda \in \mathbb{N}}$.*

In this work, we will use sub-exponentially-secure, uniform or non-uniform, collision-resistant hash functions. For $\lambda \in \mathbb{N}$ and $H \in D_\lambda$, a collision can be found by a uniform Turing machine in time $2^{\lambda/2}$ with high probability and in time 2^λ with probability 1. Furthermore, it is hard for a 2^λ-sized circuit (or a 2^λ-time uniform Turing machine) to find collisons for any hash function $H' \in D_{\lambda^{\frac{1}{\varepsilon}}}$ (or for H_n in the uniform case).

4 Basic Commitment Schemes

In this section we construct three basic over-extractable commitment schemes, each one of them enjoys hiding against different circuit classes. Firstly, we construct a depth-robust commitment scheme which is (S', S')-over-extractable and hiding against any circuit whose depth is sufficiently smaller than S'. Next, we construct a size-robust commitment scheme which is hiding against any circuit whose size is at most $\mathsf{poly}(S)$ but there exists an extractor of polynomial depth and size larger than S. Finally, we construct a commitment scheme which is hiding against both depth-restricted and size-restricted circuits.

4.1 Depth-robust Over-extractable Commitment Scheme from a TL-puzzle

For some subexponential functions T and B, assume the existence of a (T, B)-TL puzzle $(\mathsf{Gen}, \mathsf{Sol})$. For any difficulty parameter $c \log n < t(n) < B(n)$, these puzzles are solvable in time $\mathsf{poly}(2^t)$ but hard for $B(n)$-sized circuits having depth at most $\mathsf{poly}(T(t))$. [4] Furthermore, consider a difficulty parameter $t(n)$ that admits the following hierarchy of non-decreasing functions, $n << d = T(t) << S' = 2^t << S^* << B$. Using the (T, B)-TL puzzles, we construct a commitment scheme which is over-extractable in time $\mathsf{poly}(S')$ and is hiding against any circuit in $\mathcal{C}_d$ (hence the name *depth-robust* commitment scheme). We refer to the commitment scheme as $(\mathsf{ECom}_d, \mathsf{EOpen}_d)$ which is described below. [5]

On input a security parameter 1^n, the honest committer C runs the algorithm ECom_d described below to commit to a value $v \in \{0,1\}^n$. After the commit stage, the honest receiver R decides whether to accept the commitment by running the function EOpen_d as described in the reveal stage.

- Commit stage - Algorithm ECom_d:

  1. On input security parameter 1^n and value $v \in \{0,1\}^n$, for every $i \in [n]$, the honest committer C samples random strings $s_i, r_i \in \{0,1\}^n$ and computes the commitment c_i to $v[i]$, the ith bit of v, as follows,

  $$c_i = (Z_i = \mathsf{Gen}(1^n, 1^{t(n)}, s_i\ ; r),\ r_i,\ \langle r_i \cdot s_i \rangle \oplus v[i])\ ,$$

  where r is the random tape used by Gen and t is the difficulty parameter such that $d = T(t)$.

  2. C sends the vector $c = \{c_i\}_{i\in[n]}$ to R as the commitment and keeps $(v, \{s_i\}_{i\in[n]}, r)$ as the decommitment.

- Reveal stage - Function EOpen_d:
  On receiving $(v, \{s_i\}_{i\in[n]}, r)$ from C, R computes the function EOpen_d which returns 1 if $c_i = (\mathsf{Gen}(1^n, 1^t, s_i\ ; r),\ r_i,\ \langle r_i \cdot s_i \rangle \oplus v[i])$ for every $i \in [n]$. Otherwise, outputs 0.

Furthermore, the extractor $o\mathcal{E}_d$ of the scheme proceeds as follows:

- Extraction - Extractor $o\mathcal{E}_d$:
  On receiving any commitment $c = \{c_i = (Z_i, r_i, z_i)\}_{i\in[n]}$, the extractor $o\mathcal{E}_d$ computes the

[4] The definition of TL puzzles presented in Definition 17 defines hardness against circuits with depth at most T but for ease of description we assume hardness for $\mathsf{poly}(T)$ depth. This is without loss of generality for subexponential $T = 2^{t^{\delta'}}$, that is, hardness against $2^{t^{\delta'}}$ implies hardness against $\mathsf{poly}(2^{t^{\delta}})$ for any $\delta < \delta' < 1$.

[5] From now on, for notational convenience, we represent a non-interactive commitment scheme by the tuple of commit and open algorithms; that is $(\mathsf{ECom}, \mathsf{EOpen})$, instead of a pair of interactive TMs C and R.

solution s_i of Z_i by running $\mathsf{Sol}(Z_i)$. Then, $o\mathcal{E}_d$ extracts bit $v[i]$ committed in c_i by computing $v[i] = z_i \oplus \langle r_i \cdot s_i \rangle$. $o\mathcal{E}_d$ returns the string $v[0]||\ldots||v[n-1]$ as its output.

Theorem 5. *Assuming the existence of (T, B)-TL puzzle $(\mathsf{Gen}, \mathsf{Sol})$, an appropriate diffculty parameter $t(n)$ and non-decreasing functions $n << d = T(t) << S' = 2^t << S^* << \mathcal{B}$, $(\mathsf{ECom}_d, \mathsf{EOpen}_d)$ is a non-interactive, perfectly binding, $\mathcal{C}_d$-hiding, (S', S')-over-extractable commitment scheme w.r.t. extractor $o\mathcal{E}_d$.*

Proof. We discuss each of the properties in the following:

- Efficiency: For any $n \in \mathbb{N}$, difficulty parameter t which is upper-bounded by some polynomial and $i \in [n]$, ECom_d runs Gen to sample puzzles Z_i's and rest of computation (i.e., sampling n-bit strings, computing inner-product) takes $\mathsf{poly}(n)$ time. Infact for difficulty parameter $t(n)$, Gen runs in time $\mathsf{poly}(t, n)$ which is upper-bounded by some $\mathsf{poly}(n)$ as t is upper-bounded by a polynomial. Hence, ECom_d runs in time $\mathsf{poly}(n)$ for each $i \in [n]$. Therefore, ECom_d is efficient.

- Perfect binding: Note that the TL-puzzle as defined is injective, that is, given a honestly generated puzzle Z there exists only one solution s to this puzzle. Assume towards a contradiction, there exists a puzzle Z that has two solution $s_0 \neq s_1$, that is, Z lies in the support of both $\mathsf{Gen}(\cdot, \cdot, s_0)$ and $\mathsf{Gen}(\cdot, \cdot, s_1)$. Then, the deterministic algorithm Sol on input Z outputs s. If $s = s_0$, then this contradicts the correctness of Sol w.r.t. puzzles in the support of $\mathsf{Gen}(\cdot, \cdot, s_1)$ and vice-versa. Therefore, given a puzzle Z (arbitrarily generated), there exists at most one solution. This then implies that the puzzles Z_i in the commitment c lie in the support of at most one string s_i. Therefore, for every commitment c there exists at most one sequence $\{s_i\}_{i\in[n]}$ that will make R accept the commitment c. It is easy to see that this implies the perfect binding of $(\mathsf{ECom}_d, \mathsf{EOpen}_d)$.

- Over-extractable: First, the extractor $o\mathcal{E}_d$ belongs to the class $\mathcal{C}^{\wedge}_{S',S'}$ since Sol runs in time $\mathsf{poly}(S') = \mathsf{poly}(2^t)$ and the rest of the computation takes $\mathsf{poly}(n)$ time. Furthermore, since $o\mathcal{E}_d$ always solves the puzzle Z_i's correctly, it always extracts the correct unique committed value. Therefore, $(\mathsf{ECom}_d, \mathsf{EOpen}_d)$ is (S', S')-over-extractable.

- Hiding: By the definition of (T, B)-hardness of the TL puzzle, for difficulty parameter t, the distribution,
$$\{s \leftarrow \{0,1\}^n,\ Z \leftarrow \mathsf{Gen}(1^n, 1^t, s)\ :\ (s, Z)\}\ , \tag{2}$$
is unpredictable for any adversary $A = \{A_n\}_{n\in\mathbb{N}}$ where $\mathsf{dep}(A_n) \leq \mathsf{poly}(T(t))$ and $\mathsf{size}(A_n) \leq \mathsf{poly}(S^*) < B$. In our construction of $(\mathsf{ECom}_d, \mathsf{EOpen}_d)$, we sample the TL puzzles with difficulty t such that $T(t) = d$. Therefore, for any circuit in the class $\mathcal{C}_d$, the above distribution is unpredictable. We refer to such a distribution as $\mathcal{C}_d$-unpredictable. Then, by a standard argument about the hardcoreness of the Goldreich Levin bit [GL89] extracted from an $\mathcal{C}_d$-unpredictable distribution, we can conclude that the bit $\langle s_i \cdot r_i \rangle$ is hardcore for circuits in the class $\mathcal{C}_d$. This implies that $(\mathsf{ECom}_d, \mathsf{EOpen}_d)$ is $\mathcal{C}_d$-hiding.

□

4.2 Size-robust Over-extractable Commitment Scheme from OWPs

For a non-decreasing function $S(n)$ $(<< S^*(n))$, assume that there exists a OWP f that is hard to invert for any $\mathsf{poly}(S)$-sized circuit (for any polynomial $\mathsf{poly}(\cdot)$), but there exists a non-decreasing function $S''(n)$ $(S << S'' << S^*)$ such that a circuit of $\mathsf{poly}(n)$ depth and S'' size can invert

it. Such a OWP f can be instantiated from a subexponentially secure OWP by setting the input length appropriately. More concretely, consider a subexponentially secure OWP that is hard for circuits of size $\mathsf{poly}(2^{k^\varepsilon})$ (for any polynomial $\mathsf{poly}()$ and some $0 < \varepsilon < 1$). For any S, we can design the required f which is hard to invert for $\mathsf{poly}(S)$-sized circuits by setting $k = (\log S)^{1/\varepsilon}$, thereby achieving security against circuits of size $\mathsf{poly}(2^{k^\varepsilon}) = \mathsf{poly}(2^{(\log S)})$. Furthermore, there exists a circuit which can invert (with probability 1) by enumerating all the 2^k pre-images. Such a circuit has size $S'' = \mathsf{poly}(2^k) = \mathsf{poly}(2^{(\log S)^{1/\varepsilon}}) >> S$ and polynomial depth.

Using such a OWP f, we construct a commitment scheme $(\mathsf{ECom}_S, \mathsf{EOpen}_S)$ which is hiding against circuits of size $\mathsf{poly}(S)$ (hence the name *size-robust* commitment scheme) and $(\mathsf{poly}(n), S'')$-over-extractable. $(\mathsf{ECom}_S, \mathsf{EOpen}_S)$ is simply the non-interactive commitment scheme based on OWP where the hard-core predicate is the Golreich-Levin bit [GL89]. For completeness, we describe the scheme below.

- Commit stage - Algorithm ECom_S:

  1. On input security parameter 1^n and value $v \in \{0,1\}^n$, for every $i \in [n]$, the honest committer C samples random strings s_i and r_i in the domain of f and computes the commitment c_i to $v[i]$, the ith bit of v, as follows,
  $$c_i = (f(s_i),\ r_i, \langle r_i \cdot s_i \rangle \oplus v[i])\ .$$
  2. C sends the vector $c = \{c_i\}_{i\in[n]}$ to R as the commitment and keeps $(v, \{s_i\}_{i\in[n]})$ as the decommitment.

- Reveal stage - Function EOpen_S:
  On receiving $(v, \{s_i\}_{i\in[n]})$ from C, R computes the function EOpen_S which returns 1 if $c_i = (f(s_i),\ r_i,\ \langle r_i \cdot s_i \rangle \oplus v[i])$ for every $i \in [n]$. Otherwise, outputs 0.

The extractor $o\mathcal{E}_S$ for the scheme proceeds as follows:

- Extraction - Extractor $o\mathcal{E}_S$:
  On receiving any commitment $c = \{c_i = (y_i, r_i, z_i)\}_{i\in[n]}$, the extractor $o\mathcal{E}_S$ computes the pre-image s_i of y_i under f (by assumption, f can be inverted using a circuit of polynomial depth and S'' size). $o\mathcal{E}_S$ extracts bit $v[i]$ committed in c_i by computing $v[i] = z_i \oplus \langle r_i \cdot s_i \rangle$. $o\mathcal{E}_S$ returns the string $v[0]||\ldots||v[n-1]$ as its output.

Theorem 6. *If f is a $\mathcal{C}_S$-secure OWP which is invertible by a circuit in $\mathcal{C}^{\wedge}_{\mathsf{poly},S''}$ for some $S'' >> S$ then $(\mathsf{ECom}_S, \mathsf{EOpen}_S)$ is a non-interactive, perfectly binding, $\mathcal{C}_S$-hiding and (poly, S'')-over-extractable commitment scheme w.r.t. extractor $o\mathcal{E}_S$.*

Proof. We discuss all the properties in the following:

- Binding and Hiding: The proof of perfect binding follows from the injectivity of f and proof of $\mathcal{C}_S$-hiding follows from the hard-coreness of the Goldreich-Levin bit with OWP being $\mathcal{C}_S$-secure (hence the scheme is $\mathcal{C}_S$-hiding).

- Over-extractable: First, the extractor $o\mathcal{E}_S$ belongs to the class $\mathcal{C}^{\wedge}_{\mathsf{poly},S''}$ since f can be inverted by a circuit in $\mathcal{C}^{\wedge}_{\mathsf{poly},S''}$ and the rest of the computation takes $\mathsf{poly}(n)$ time. Furthermore, since $o\mathcal{E}_S$ always inverts the OWP images y_i's correctly, it always extracts the correct unique committed value. Therefore, $(\mathsf{ECom}_S, \mathsf{EOpen}_S)$ is (poly, S'')-over-extractable.

□

4.3 Strong Over-extractable Commitment Scheme

For non-decreasing functions,

$$n << d(n) << S'(n), S(n) << S''(n) << S^*(n) << 2^{n^\varepsilon} \ ,$$

we construct a non-interactive perfectly binding commitment scheme $(\mathsf{ECom}_{d,S}, \mathsf{EOpen}_{d,S})$ which is $\mathcal{C}^{\vee}_{d,S}$-hiding and (S', S'')-over-extractable w.r.t an extractor $o\mathcal{E}_{d,S}$. Note that, unlike the commitment schemes described in Sections 4.1 and 4.2 which were either hiding against depth-restricted circuits $\mathcal{C}_d$ or hiding against size-restricted circuits $\mathcal{C}_S$, $(\mathsf{ECom}_{d,S}, \mathsf{EOpen}_{d,S})$ enjoys a *stronger* security property of being hiding against circuits in both depth-restricted and size-restricted circuit classes (i.e., $\mathcal{C}^{\vee}_{d,S}$). We describe the construction of the scheme $(\mathsf{ECom}_{d,S}, \mathsf{EOpen}_{d,S})$ for an honest committer C and an honest receiver R below. The idea is to commit to a random 2-out-of-2 secret share of the value v using each of the schemes described in Sections 4.1 and 4.2.

- Commit stage - Algorithm $\mathsf{ECom}_{d,S}$:

  1. On input security parameter 1^n and value $v \in \{0,1\}^n$, C samples a random n-bit string r_0.
  2. C computes a commitment c_1 to r_0 using ECom_d. Let d_1 be the corresponding decommitment string.
  3. C computes a commitment c_2 to $v \oplus r_0$ using ECom_S. Let d_2 be the corresponding decommitment string.
  4. C sends (c_1, c_2) as the commitment c to R and keeps the decommitment (v, r_0, d_1, d_2) private.

- Reveal stage - Function $\mathsf{EOpen}_{d,S}$:
  On receiving the decommitment (v, r_0, d_1, d_2), R accepts it if both EOpen_d and EOpen_S accept the corresponding decommitments; that is,

  $$\mathsf{EOpen}_d(c_1, r_0, d_1) = 1 \wedge \mathsf{EOpen}_S(c_2, v \oplus r_0, d_2) = 1 \ .$$

  Otherwise, R rejects.

The extractor $o\mathcal{E}_{d,S}$ of the scheme proceeds as follows:

- Extraction - Extractor $o\mathcal{E}_{d,S}$:
  The extractor $o\mathcal{E}_{d,S}$ on input $c = (c_1, c_2)$ runs the extractors $o\mathcal{E}_d$ and $o\mathcal{E}_S$ with inputs c_1 and c_2, obtaining outputs r'_0 and r'_1 respectively. If either r'_0 or r'_1 is $\perp$ then $o\mathcal{E}_{d,S}$ outputs $\perp$. Otherwise, $o\mathcal{E}_{d,S}$ outputs $r'_0 \oplus r'_1$.

Theorem 7. *$(\mathsf{ECom}_{d,S}, \mathsf{EOpen}_{d,S})$ is a non-interactive, perfectly binding, $\mathcal{C}^{\vee}_{d,S}$-hiding and (S', S'')-over-extractable commitment scheme w.r.t. extractor $o\mathcal{E}_{d,S}$.*

Proof. We discuss each of the properties in the following:

- Perfect binding: The perfect binding follows from the perfect binding of ECom_d and ECom_S.

- Over-extractable: A valid commitment $c = (c_1, c_2)$ is such that both c_1 and c_2 are valid commitments for ECom_d and ECom_S respectively. Since ECom_d and ECom_S are over-extractable w.r.t. extractors $o\mathcal{E}_d$ and $o\mathcal{E}_S$ respectively, $o\mathcal{E}_{d,S}$ which runs $o\mathcal{E}_d(c_1)$ and $o\mathcal{E}_S(c_2)$ extracts out the unique committed values and hence outputs $\mathsf{val}(c)$ with over-whelming probability. Furthemore, $o\mathcal{E}_d \in \mathcal{C}^{\wedge}_{S',S'}$ and $o\mathcal{E}_S \in \mathcal{C}^{\wedge}_{\mathsf{poly},S''}$ implies that $o\mathcal{E}_{d,S}$ belongs to the circuit class $\mathcal{C}^{\wedge}_{S',S''}$.

- Hiding: Assume towards a contradiction that there exists a non-uniform circuit family $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\vee}_{d,S}$, and for some polynomial $p(\cdot)$ and infinitely many $n \in \mathbb{N}$, a pair of values $v_0, v_1 \in \{0,1\}^n$,

$$\mathsf{Pr}\left[b \leftarrow \{0,1\}, c \leftarrow \mathsf{ECom}_{d,S}(1^n, v_b) : b = A_n(c)\right] \geq \frac{1}{2} + \frac{1}{p(n)} \ . \tag{3}$$

  Using A, we construct a non-uniform circuit family $B = \{B_n\}_{n\in\mathbb{N}}$ that breaks the hiding of either ECom_d or ECom_S depending on the depth and size of A. Since $A \in \mathcal{C}^{\vee}_{d,S}$, it could either be that $A \in \mathcal{C}_d$ or $A \in \mathcal{C}_S$. We will consider the two cases separately below.

  Case 1 - $A \in \mathcal{C}_S$: In this case, we construct a B that violates the hiding of ECom_S as follows: B_n with v_0 and v_1 hard-wired in it, samples a random n-bit string r_0 and computes a commitment c_1 to string r_0 using ECom_d. It sends $(v_0 \oplus r_0)$ and $(v_1 \oplus r_0)$ as challenges in the hiding game of ECom_S and receives a commitment c_2 to $(v_b \oplus r_0)$, for a randomly chosen bit b. Finally, B_n sends the tuple (c_1, c_2) as the commitment to A_n and forwards the output of A_n as its output. B perfectly simulates the hiding game of $\mathsf{ECom}_{d,S}$ for A while itself participating in the hiding game of ECom_S and hence succeeds with probability at least $\frac{1}{2} + \frac{1}{p(n)}$. Furthermore, since B incurs only polynomial blow-up in size over A (while simulating the game for A), we have $B \in \mathcal{C}_S$. Therefore, $B \in \mathcal{C}_S$ succeeds in the hiding game of ECom_S with non-negligible probability away from $\frac{1}{2}$, which is a contradiction.

  Case 2 - $A \in \mathcal{C}_d$: The proof for Case 2 is similar to Case 1 but here we, instead, construct $B \in \mathcal{C}_d$ which succeeds in the hiding game of ECom_d with non-negligible probability away from $\frac{1}{2}$. The only difference from the previous case is that B commits to r_0 using the scheme ECom_S and forwards $(v_0 \oplus r_0)$ and $(v_1 \oplus r_0)$ as challenges in the hiding game of ECom_d. Since the marginal distribution of both random shares of v (i.e., r and $v \oplus r$ for a random r) are identical, B still perfectly simulates the hiding game of $\mathsf{ECom}_{d,S}$ for A.

□

5 Non-malleable Commitment Scheme w.r.t. Extraction for Short Identities

Assume that we have the following hierarchy of non-decreasing functions on $\mathbb{N}$,

$$n << d_0 << d_1 << \ldots << d_{l-1} << d_l << S_0 << S_1 << \ldots << S_{l-1} << S_l << S^* << 2^{n^\varepsilon} \ , \tag{4}$$

such that for every $\mathsf{id} \in [l]$,

- there exists a depth-robust commitment scheme $(\mathsf{ECom}_{d_{\mathsf{id}}}, \mathsf{EOpen}_{d_{\mathsf{id}}})$ that is $\mathcal{C}_{d_{\mathsf{id}}}$-hiding and $(d_{\mathsf{id}+1}, d_{\mathsf{id}+1})$-over-extractable w.r.t. an extractor $o\mathcal{E}_{d_{\mathsf{id}}}$.

- there exists a size-robust commitment scheme $(\mathsf{ECom}_{S_{\mathsf{id}}}, \mathsf{EOpen}_{S_{\mathsf{id}}})$ that is $\mathcal{C}_{S_{\mathsf{id}}}$-hiding and $(\mathsf{poly}(n), S_{\mathsf{id}+1})$-over-extractable w.r.t. an extractor $o\mathcal{E}_{S_{\mathsf{id}}}$.

By Section 4.3, we can then construct a family of l commitment schemes $\{(\mathsf{ECom}_{\mathsf{id}}, \mathsf{EOpen}_{\mathsf{id}})\}_{\mathsf{id}\in[l]}$ such that for every $\mathsf{id} \in [l]$,

$$(\mathsf{ECom}_{\mathsf{id}}, \mathsf{EOpen}_{\mathsf{id}}) = (\mathsf{ECom}_{d_{\mathsf{id}}, S_{l-\mathsf{id}-1}}, \mathsf{EOpen}_{d_{\mathsf{id}}, S_{l-\mathsf{id}-1}}) \ ,$$

and by Theorem 7 we have that $(\mathsf{ECom}_{\mathsf{id}}, \mathsf{EOpen}_{\mathsf{id}})$ is a non-interactive, perfectly binding, $\mathcal{C}^{\vee}_{d_{\mathsf{id}}, S_{l-\mathsf{id}-1}}$-hiding and $(d_{\mathsf{id}+1}, S_{l-\mathsf{id}})$-over-extractable commitment scheme w.r.t. an extractor $o\mathcal{E}_{\mathsf{id}}$ (described in Section 4.3). We use this family of l commitment schemes to construct a tag-based commitment scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ for identities of length $\log l$-bits which is one-one non-malleable w.r.t. extraction by an extractor $o\mathcal{E}_{\mathsf{NM}}$. We describe the scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ and the extractor $o\mathcal{E}_{\mathsf{NM}}$ below.

- Commit stage - Algorithm ENMCom:

  1. On input security parameter 1^n, identity $\mathsf{id} \in [l]$ and a value $v \in \{0,1\}^n$, C computes a commitment c to v using $\mathsf{ECom}_{\mathsf{id}}$. Let d be the corresponding decommitment string.
  2. C sends the commitment c to R and keeps the decommitment (v, d) private.

- Reveal stage - Function $\mathsf{ENMOpen}$:
  On receiving the decommitment (v, d) and identity id, R computes $\mathsf{ENMOpen}(\mathsf{id}, c, v, d)$ which returns 1 if $\mathsf{EOpen}_{\mathsf{id}}(c, v, d)$ returns 1. Otherwise, returns 0.

The extractor $o\mathcal{E}_{\mathsf{NM}}$ proceeds as follows,

- Extraction - Extractor $o\mathcal{E}_{\mathsf{NM}}$:
  The extractor $o\mathcal{E}_{\mathsf{NM}}$ on input c and identity id outputs the value extracted by $o\mathcal{E}_{\mathsf{id}}$ from c.

Remark 3. *We want* ENMCom *and* $\mathsf{ENMOpen}$ *to be computable by uniform TMs. This mandates that* $\{\mathsf{ECom}_{\mathsf{id}}\}_{\mathsf{id}\in[l]}$ *and* $\{\mathsf{EOpen}_{\mathsf{id}}\}_{\mathsf{id}\in[l]}$ *be uniformly and efficiently computable; that is, there must exist uniform* PPT *TMs* M_{com} *and* M_{open} *that on input* id *can compute* $\mathsf{ECom}_{\mathsf{id}}$ *and* $\mathsf{EOpen}_{\mathsf{id}}$ *respectively. If* $l = O(1)$ *then one can simply hard-code all the algorithms* $\{\mathsf{ECom}_{\mathsf{id}}\}_{\mathsf{id}\in[l]}$ *and* $\{\mathsf{EOpen}_{\mathsf{id}}\}_{\mathsf{id}\in[l]}$ *in* M_{com} *and* M_{open} *respectively. As will see later,* $l = O(1)$ *is sufficient for constructing non-malleable commitment scheme for* n*-bit identities. When* $l = \omega(1)$ *the hard-coding approach, in fact, does not work. Nevertheless, we note that the algorithms* $\mathsf{ECom}_{\mathsf{id}}$ *and* $\mathsf{EOpen}_{\mathsf{id}}$ *described in Section 4.3 are still efficiently and uniformly computable. Since, this case does not occur in our construction, we omit details here.*

Theorem 8. $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ *is a non-interactive, perfectly binding,* $\mathcal{C}^{\wedge}_{d_0, S_0}$*-hiding,* (d_l, S_l)*-over-extractable tag-based commitment scheme for identities of length* $\log l$*. Furthermore,* $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ *is one-one* $\mathcal{C}^{\wedge}_{d_0, S_0}$*-non-malleable w.r.t. extraction by extractor* $o\mathcal{E}_{\mathsf{NM}}$*.*

We note that both hiding and non-malleability hold only against circuits in the restrictive class $\mathcal{C}^{\wedge}_{d_0, S_0}$; that is, circuits A whose depth and size are bounded by $\mathsf{poly}(d_0)$ and $\mathsf{poly}(S_0)$ respectively, even though the building blocks $\mathsf{ECom}_{\mathsf{id}}$'s have the stronger security of being hiding against circuits in $\mathcal{C}^{\vee}_{d_{\mathsf{id}}, S_{l-\mathsf{id}-1}} \supset \mathcal{C}^{\wedge}_{d_0, S_0}$; that is, circuits A which are either restricted in their depths or their size but not both.

Proof. The perfect binding follows readily from the perfect binding of each of the $\mathsf{ECom}_{\mathsf{id}}$'s. We discuss over-extractability and non-malleability in the following:

- Over-extractable: A valid commitment c with identity id is a valid commitment for $\mathsf{ECom}_{\mathsf{id}}$. Therefore, the extractor $o\mathcal{E}_{\mathsf{NM}}$ which runs $o\mathcal{E}_{\mathsf{id}}$ on c extracts the correct unique committed value due to the over-extractability of $\mathsf{ECom}_{\mathsf{id}}$ w.r.t. $o\mathcal{E}_{\mathsf{id}}$. Furthermore, $\mathsf{ECom}_{\mathsf{id}}$'s are $(d_{\mathsf{id}+1}, S_{l-\mathsf{id}})$-over-extractable and hence the depth of $o\mathcal{E}_{\mathsf{id}}$ is at most $\mathsf{poly}(d_{\mathsf{id}+1})$ and size is at most $\mathsf{poly}(S_{l-\mathsf{id}})$. Therefore, $o\mathcal{E}_{\mathsf{NM}}$ (which runs $o\mathcal{E}_{\mathsf{id}}$) is a circuit with depth bounded by $\mathsf{poly}(d_l)$ and size bounded by $\mathsf{poly}(S_l)$ (see Inequality 4). Hence, $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ is (d_l, S_l)-over-extractable.

- Non-malleability and Hiding: The proof of hiding follows from the proof of non-malleability (described below). For proving one-one non-malleability w.r.t. extraction by $o\mathcal{E}_{\mathsf{NM}}$, let us assume for contradiction that there exists a non-uniform circuit $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\wedge}_{d_0,S_0}$ which participates in one left and one right interaction such that for infinitely many $n \in \mathbb{N}$ there exists $v_0, v_1 \in \{0,1\}^n$ such that the following distributions are computationally distinguishable,

$$\mathsf{emim}^{A}_{\mathsf{ENMCom}}(v_0) \; ; \mathsf{emim}^{A}_{\mathsf{ENMCom}}(v_1) \; . \tag{5}$$

  Equivalently, there exists a non-uniform circuit $D = \{D_n\}_{n\in\mathbb{N}} \in \mathcal{P}/\mathrm{poly}$ and a polynomial $p(\cdot)$ such that D distinguishes the above distributions with non-negligible advantage $\frac{1}{p(n)}$. Let id and $\tilde{\mathsf{id}}$ be the identities chosen by A in the left and right interactions respectively. Note that since the only message A receives in the execution is the left commitment and identity for the left interaction needs to be chosen before that, we can assume that the left side identity id is fixed.

  Using A and D, we will construct a non-uniform circuit $B = \{B_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}-1}}$ that breaks the hiding of $\mathsf{ECom}_{\mathsf{id}}$ with advantage at least $\frac{1}{p(n)}$. More concretely, B internally runs A and acts as an honest committer in the left interaction with A while as an honest receiver in the right interaction. In the hiding game of $\mathsf{ECom}_{\mathsf{id}}$, B sends (v_0, v_1) as challenges and receives a commitment c to v_b, for a randomly chosen bit b. B forwards c to A as the commitment in the left interaction. A sends a commitment $\tilde{c}$ to the honest right receiver (simulated by B). Then, B runs the extractor $o\mathcal{E}_{\tilde{\mathsf{id}}}$ on $\tilde{c}$ obtaining an extracted value $\tilde{v}'$. Depending on the value of b, the over-extracted value $\tilde{v}'$ along with the view of A is identical to $\mathsf{emim}^{A}_{\mathsf{ENMCom}}(v_b)$. B runs the distinguisher D with inputs $\tilde{v}'$ and the view of A. Finally, B returns the output of D as its output.

  By our hypothesis, B succeeds in breaking the hiding of $\mathsf{ECom}_{\mathsf{id}}$ with advantage at least $\frac{1}{p(n)}$. Now to arrive at a contradiction it remains to show that $B \in \mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}-1}}$. B runs the extractor $o\mathcal{E}_{\tilde{\mathsf{id}}} \in \mathcal{C}^{\wedge}_{d_{\mathsf{id}+1},S_{l-\tilde{\mathsf{id}}}}$ and $A \in \mathcal{C}^{\wedge}_{d_0,S_0}$, while the rest of the simulation takes $\mathsf{poly}(n)$ time. Therefore the depth of B is such that,

$$\begin{aligned} \mathsf{dep}(B) &= \mathsf{dep}(A) + \mathsf{dep}(o\mathcal{E}_{\tilde{\mathsf{id}}}) + \mathsf{poly}(n) \\ &\leq \mathsf{poly}(d_0) + \mathsf{poly}(d_{\tilde{\mathsf{id}}+1}) + \mathsf{poly}(n) < \mathsf{poly}(d_{\tilde{\mathsf{id}}+1}) \; . \end{aligned} \tag{6}$$

  Similarly, the size of B is such that,

$$\begin{aligned} \mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(o\mathcal{E}_{\tilde{\mathsf{id}}}) + \mathsf{poly}(n) \\ &\leq \mathsf{poly}(S_0) + \mathsf{poly}(S_{l-\tilde{\mathsf{id}}}) + \mathsf{poly}(n) \\ &< \mathsf{poly}(S_{l-\tilde{\mathsf{id}}}) << S^* \; . \end{aligned} \tag{7}$$

We consider two cases for the identities id and $\tilde{\mathsf{id}}$ as follows: [6]

Case 1 - $\mathsf{id} > \tilde{\mathsf{id}}$: In this case, $d_{\mathsf{id}} \geq d_{\tilde{\mathsf{id}}+1}$, we have that $\mathsf{dep}(B) < \mathsf{poly}(d_{\mathsf{id}})$ for some polynomial $\mathsf{poly}(\cdot)$. Therefore, $B \in \mathcal{C}_{d_{\mathsf{id}}}$ and hence $B \in \mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}-1}}$.

Case 2 - $\mathsf{id} < \tilde{\mathsf{id}}$: In this case, $S_{l-\tilde{\mathsf{id}}} \leq S_{l-\mathsf{id}-1}$ we have that $\mathsf{size}(B) < \mathsf{poly}(S_{l-\mathsf{id}-1})$ for some polynomial $\mathsf{poly}(\cdot)$. Therefore $B \in \mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}-1}}$.

Thus, irrespective of the identity $\tilde{\mathsf{id}}$ chosen by A for the right interaction, we can construct $B \in \mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}-1}}$ which breaks hiding of $\mathsf{ECom}_{\mathsf{id}}$ with non-negligible advantage, which is a contradiction.

□

Remark 4. *In the above proof, the reduction B which bases the one-one non-malleability w.r.t. extraction on the hiding of $\mathsf{ECom}_{\mathsf{id}}$, runs both A and the extractor $o\mathcal{E}_{\tilde{\mathsf{id}}}$ of the commitment scheme $\mathsf{ECom}_{\tilde{\mathsf{id}}}$. Therefore, B has depth at most $\mathsf{dep}(A) + \mathsf{poly}(d_{\tilde{\mathsf{id}}+1})$ and has size at most $\mathsf{size}(A) + \mathsf{poly}(S_{l-\tilde{\mathsf{id}}})$ respectively. To reach a contradiction, one must argue that the reduction B belongs to $\mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}}}$. In other words, either $\mathsf{dep}(A) + \mathsf{poly}(d_{\tilde{\mathsf{id}}+1})$ is at most $\mathsf{poly}(d_{\mathsf{id}})$ or $\mathsf{size}(A) + \mathsf{poly}(S_{l-\tilde{\mathsf{id}}})$ is at most $\mathsf{poly}(S_{l-\mathsf{id}-1})$. Since A chooses both id and $\tilde{\mathsf{id}}$, this can only hold if $\mathsf{dep}(A)$ and $\mathsf{size}(A)$ are both small; that is, $o(d_1)$ and $o(S_1)$ respectively. As a result, we only show non-malleability of $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ against weak adversaries whose depth and size both are bounded by $\mathsf{poly}(d_0) = o(d_1)$ and $\mathsf{poly}(S_0) = o(S_1)$ respectively.*

Remark 5. *Furthermore, we note that even though $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ is non-malleable w.r.t. extraction, we cannot prove that it is non-malleable (w.r.t. commitment). This is because the underlying commitment schemes $\mathsf{ECom}_{\mathsf{id}}$'s are only over-extractable. Over-extractability guarantees that for a valid commitment, the value extracted by the extractor is indeed the value committed (except with negligible probability). However, when a commitment is invalid, the extracted value can be arbitrary – hence the name over-extractable. Therefore, there might exist an adversary A that depending on the value committed on the left sends invalid commitments with different probabilities on the right. Such an adversary clearly violates the non-malleability (w.r.t. commitment) but may not violate non-malleability w.r.t. extraction. This is because the over-extracted values may still be indistinguishable. Hence, we cannot base non-malleability (w.r.t. commitment) on non-malleability w.r.t. extraction of $(\mathsf{ENMCom}, \mathsf{ENMOpen})$.*

6 Strengthening Non-malleability

The commitment scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ described in Section 5 is only stand-alone (one-one) non-malleable w.r.t. extraction. However, our final goal is to construct a scheme that is concurrent non-malleable (w.r.t. commitment). In this section, we describe a transformation that transforms any 2-round commitment scheme $\langle C, R \rangle$ which is one-one non-malleable w.r.t. extraction into a 2-round commitment scheme $\langle \widehat{C}, \widehat{R} \rangle$ which is concurrent non-malleable w.r.t. extraction as well as concurrent non-malleable (w.r.t. commitment), while preserving the length of the identities. Below, we first describe a bare-bone protocol that has two main problems; we discuss how to resolve them, which naturally leads to our transformation.

[6] Note that the case $\mathsf{id} = \tilde{\mathsf{id}}$ is not invalid execution and hence not considered.

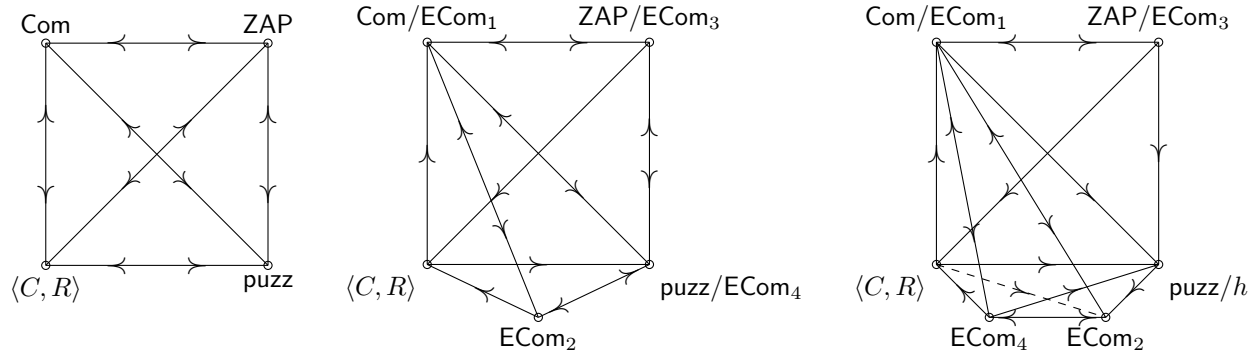
6.1 A Bare-Bone Protocol and Challenges

As discussed in the overview in Section 2, our construction of $\langle \widehat{C}, \widehat{R} \rangle$ is inspired by the non-malleability amplification technique in [LP09]. As a starting point, their technique suggests the following bare-bone protocol:

A Bare-Bone Protocol $\langle \widehat{C}, \widehat{R} \rangle$: The receiver sends a *puzzle* puzz, together with the first message a_{NM} of $\langle C, R \rangle$ and the first message a_{ZAP} of ZAP. The commiter commits to v using a non-interactive commitment scheme Com, sends the second message b_{NM} of $\langle C, R \rangle$ committing to a random string $r1$, and the second message b_{ZAP} of ZAP proving that either i) $c1$ commits to v or ii) $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to a solution s of the puzzle puzz (which is efficiently verifiable).

$$\widehat{C} \quad \xleftarrow{\mathsf{puzz},\ a_{\mathsf{NM}},\ a_{\mathsf{ZAP}}} \quad \widehat{R}$$
$$\widehat{C} \quad \xrightarrow{\mathsf{Com(v)},\ b_{\mathsf{NM}},\ b_{\mathsf{ZAP}}} \quad \widehat{R}$$

As discussed before, to show the security of such a bare-bone protocol, *ideally*, we would like different components — puzz, $\langle C, R \rangle$, Com, and ZAP — to be *mutually non-malleable*. Informally speaking, we say that a primitive P is more secure than a primitive Q, denoted as $\mathsf{P} \succ \mathsf{Q}$, if the security of P holds even when security of Q is broken by force; P and Q are mutually non-malleable if $\mathsf{P} \prec\succ \mathsf{Q}$. The ideal configuration is illustrated in Figure 2 (i). Towards realizing as many constraints in the ideal configuration as possible, the first idea is using three size-and-depth robust commitment schemes $\mathsf{ECom}_1, \mathsf{ECom}_4, \mathsf{ECom}_3$ [7] to implement Com and puzz, and augment ZAP so that they become mutually non-malleable. But, we run into problems with respect to the input non-malleable commitment $\langle C, R \rangle$.



(i) Ideal Configuration (ii) Assume NIWI (iii) Assume CRH

Figure 2: The relation between different primitives. **(i)**: The ideal configuration where all primitives are mutually non-malleable to each other; however, it cannot be instantiated. **(ii)** A sufficient configuration; it can be instantiated assuming NIWI. **(iii)**: A sufficient configuration, which can be instantiated assuming collision resistant hash functions or one-way permutations. (The dashed line is by transitivity.)

Challenge 1: $\langle C, R \rangle$ is only secure against adversaries which have both bounded depth *AND* bounded size. (Technically, it is secure against $\mathcal{C}^{\wedge}_{d_{\mathsf{NM}}, S_{\mathsf{NM}}}$, for some d_{NM} and S_{NM}; this is the

[7] The indexes are as such in order to match the protocol description later.

case for the basic schemes constructed in Section 5, as well as the schemes produced by the transformation in this section.) This type of *AND* security means either a primitive P is more secure than $\langle C, R\rangle$ or less, but cannot be mutually non-malleable. Though through a more careful analysis, we can remove some constraints w.r.t. the non-malleable commitment, it still requires $\langle C, R\rangle \prec\!\succ \mathsf{puzz}$, in order to show the security of the bare-bone protocol.

Challenge 2: In addition, constructing a puzzle from size-and-depth robust commtiment ECom_4 is not straightforward. If we naively use $\mathsf{puzz} = \mathsf{ECom}_4(s)$ as a puzzle, a malicious man-in-the-middle can send an invalid commitment, which has no solution; this would make the security proof stuck. To prevent this, one straightforward approach is asking the receiver to send two puzzles and prove using NIWI that at least one of them is well-formed. However, this requires relying the exsitence of NIWI.

To resolve Challenge 1, we modify the bare bone protocol using an additional size-and-robust commitment ECom_2. The key idea is creating a "buffer" between $\langle C, R\rangle$ and puzz, by setting the following relation: $\mathsf{ECom}_2 \succ \langle C, R\rangle$, $\langle C, R\rangle \succ \mathsf{puzz}$, and $\mathsf{ECom}_2 \prec\!\succ \mathsf{puzz}$, as illustrated in Figure 2 (ii). Note that now the non-malleable commitment does not need to satisfy mutual non-malleability with either ECom_2 or puzz. On the other hand, the mutual non-malleability of ECom_2 and puzz helps the security proof to go through.

However, to fulfill the relation $\mathsf{ECom}_2 \prec\!\succ \mathsf{puzz}$, it seems necessary to instantiate puzz using a size-and-depth robust commitment scheme, which however would involve using NIWI. To avoid this, we would like to set puzz to be, for example, a randomly chosen collision resistant hash (CRH) function h, or a randomly chosen image $y = f(s)$ of a one-way permutation (OWP), whose corresponding solutions are respectively a collision of h and a preimage of y. These puzzles have the advantage that their validity are efficiently verifiable and hence NIWI can be disposed. But, a problem with using, say, h as the puzzle is that, it cannot be mutually non-malleable with ECom_2. To resolve this, we use a $h \succ \mathsf{ECom}_2$, and to compensate for the fact that $h \not\succ \mathsf{ECom}_2$, we use non-uniformity in the proof as follows: When reducing to the security of ECom_2, the reduction instead of finding a collisioin of h by force, receives a collision as a non-uniform advice. This can be done since the puzzle h is sent in the first message completely before the ECom_2 commitment.

Unfortunately, instantiating the puzzles using CRH or OWP creates another problem: Given that $\langle C, R\rangle \succ \mathsf{puzz} = h$ and $h \succ \mathsf{ECom}_2$, it actually implies that $\langle C, R\rangle \succ \mathsf{ECom}_2$. This transitivity holds because the h is only secure against attackers with bounded size. (If h were replaced with another size-and-depth robust commitment ECom', then transitivity does not hold in general.) But this means $\langle C, R\rangle$ needs to be mutually non-malleable with ECom_2 again. To solve this problem, we again use the idea of creating "buffers". More specifically, we set the following relation: $\mathsf{ECom}_4 \succ \langle C, R\rangle$, $\langle C, R\rangle \succ \mathsf{puzz}$, $\mathsf{puzz} \succ \mathsf{ECom}_2$, and $\mathsf{ECom}_2 \prec\!\succ \mathsf{ECom}_4$, as illustrated in Figure 2 (iii). Now transitivity implies that $\langle C, R\rangle \succ \mathsf{ECom}_2$, but $\langle C, R\rangle$ no longer need to be simultanously weaker than ECom_2, and only needs to be weaker than the new "buffer" ECom_4. Moreover, the mutual non-malleability between ECom_2 and ECom_4 helps the proof to go through.

6.2 Building Blocks

Our transformation will make use of the following building blocks. We note that the parameters associated with these building blocks are set so as to satisfy the above discussed relations as in Figure 2 (iii).

For some hierarchy of non-decreasing functions on $\mathbb{N}$ satisfying,

$$\begin{aligned} &n << d_4 << d_3 << d_2 << d_1 << S_2 << S_{\mathsf{CRH}} << \\ &S'_{\mathsf{CRH}} << S_{\mathsf{NM}} << S'_{\mathsf{NM}} << S_3 << S_4 << S'_4 << S^* \ , \end{aligned} \tag{8}$$

the transformation relies on the following building blocks,

1. $\langle C, R\rangle$ is a 2-round, tag-based commitment scheme for $t(n)$-bit identities that is $(S'_{\mathsf{NM}}, S'_{\mathsf{NM}})$-over-extractable by extractor $o\mathcal{E}_{\mathsf{NM}}$. Furthermore, $\langle C, R\rangle$ is one-one $\mathcal{C}^{\wedge}_{S_{\mathsf{NM}},S_{\mathsf{NM}}}$-non-malleable w.r.t. extraction by $o\mathcal{E}_{\mathsf{NM}}$. [8]

2. $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ is a perfectly binding depth-robust commitment scheme which is $\mathcal{C}_{d_1}$-hiding and (S_2, S_2)-over-extractable w.r.t. extractor $o\mathcal{E}_1$.

3. $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_2,S_2}$-hiding and (d_1, S_{CRH})-over-extractable w.r.t. extractor $o\mathcal{E}_2$.

4. $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_3,S_3}$-hiding and (d_2, S_4)-over-extractable w.r.t. extractor $o\mathcal{E}_3$.

5. $(\mathsf{ECom}_4, \mathsf{ECom}_4)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_4,S_4}$-hiding and (d_3, S'_4)-over-extractable w.r.t. extractor $o\mathcal{E}_4$.

6. ZAP is a 2-round $\mathcal{C}_{S^*}$-witness-indistinguishable proof.

7. $\mathcal{H} = \{D_n\}$ is a family of non-uniform $\mathcal{C}_{S_{\mathsf{CRH}}}$-collision resistant hash functions such that there exists a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$ which finds collisions for $\mathcal{H}$ with probability 1.

6.3 Commitment Scheme $\langle \widehat{C}, \widehat{R}\rangle$

Using buiding blocks described in the previous subsection, we now describe our construction of a 2-round, tag-based commitment scheme $\langle \widehat{C}, \widehat{R}\rangle$ for $t(n)$-bit identities that is (S_2, S_2)-over-extractable w.r.t. an extractor $\widehat{o\mathcal{E}}_{\mathsf{NM}}$, and show that it is both concurrent $\mathcal{C}^{\wedge}_{d_4,d_4}$-non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ and concurrent $\mathcal{C}^{\wedge}_{d_4,d_4}$-non-malleable (w.r.t. commitment).

The committer $\widehat{C}$ and the receiver $\widehat{R}$ receive the security parameter 1^n and identity $\mathsf{id} \in \{0,1\}^{t(n)}$ as common input. Furthermore, $\widehat{C}$ gets a private input $v \in \{0,1\}^n$ which is the value to be committed.

- Commit stage - First round:

  1. $\widehat{R}$ samples a hash function h from $\mathcal{H}$.
  2. $\widehat{R}$ samples the first message a_{ZAP} of ZAP.
  3. $\widehat{R}$ generates the first message a_{NM} of $\langle C, R\rangle$ using the honest receiver R with identity id.
  4. $\widehat{R}$ sends $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ as the first round message to $\widehat{C}$.

- Commit stage - Second round:

[8] The non-interactive scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ of Section 5 can be viewed as a 2-round scheme $\langle C, R\rangle$ where the first round message from R is the null string.

1. (a) $\widehat{C}$ computes a commitment $c1$ to the value v using ECom_1. Let $d1$ be the corresponding decommitment string.
   (b) $\widehat{C}$ computes a commitment $c3$ to the decommitment $(v, d1)$ of $c1$ using ECom_3.
2. (a) $\widehat{C}$ computes a commitment $c2$ to a random string $r1$ using ECom_2.
   (b) Given a_{NM}, $\widehat{C}$ computes the second message b_{NM} of $\langle C, R\rangle$ using the honest committer C with identity id to commit to a random string $r2$.
   (c) $\widehat{C}$ computes a commitment $c4$ to a random string $r3$ using ECom_4.
3. Given a_{ZAP}, $\widehat{C}$ computes the second message b_{ZAP} of ZAP to prove the following OR-statement:
   (a) *either* there exists a string $\bar{v}$ such that $c1$ is a commitment to $\bar{v}$ and $c3$ commits to a decommitment of $c1$.
   (b) *or* there exists a string $\bar{s} = (x_1, x_2)$ such that $c2$ is a commitment to $\bar{s}$ and $c4$ commits to a decommitment of $c2$ and $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commit to a decommitment of $c4$ and $h(x_1) = h(x_2)$.

   $\widehat{C}$ proves the statement (a) by using a decommitment of $c3$ to $(v, d1)$ — decommitment of $c1$ to v — as the witness.
4. $\widehat{C}$ sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as the second message to $\widehat{R}$ and keeps the decommitment $(v, d1)$ private.

- Reveal stage:
  On receiving $(v, d1)$ from $\widehat{C}$, $\widehat{R}$ accepts the decommitment if the ZAP proof is accepting and if $\mathsf{EOpen}_1(c1, v, d1) = 1$. Otherwise, it rejects.

We refer to the entire transcript of the interaction as the commitment c. Moreover, we say that an interaction (with transcript c) is *accepting* if the ZAP proof contained in the commitment c is accepting. According to the reveal stage, the value of a commitment c, $\mathsf{val}(c)$ is the value committed under $c1$ (contained in c) if c is accepting. Otherwise, $\mathsf{val}(c)$ is $\perp$.

Next, we describe the extractor $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ of the scheme below.

- Extraction - Extractor $\widehat{o\mathcal{E}_{\mathsf{NM}}}$:
  On receiving a commitment c and identity id, $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ first verifies the ZAP proof and outputs $\perp$ if the proof is not accepting. Otherwise, it runs the extractor $o\mathcal{E}_1$ on $c1$ and outputs the extracted value v'.

Theorem 9. *$\langle \widehat{C}, \widehat{R}\rangle$ is a 2-round, perfectly binding, $\mathcal{C}^{\wedge}_{d_4,d_4}$-hiding, (S_2, S_2)-over-extractable commitment scheme for identities of length $t(n)$.*

Proof. The perfectly binding property follows from that of the non-interactive commitment scheme $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$. The proof of hiding will follow from the proof of Theorem 10, which we present later.

- Over-extractability: A valid commitment c to a value v, from the definition of reveal stage of $\langle \widehat{C}, \widehat{R}\rangle$, is such that the ZAP proof contained in c is accepting and $c1$ (contained in c) is a valid commitment to v using ECom_1. In this case, the extractor $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ runs $o\mathcal{E}_1$ on $c1$, which by the over-extractability of ECom_1 w.r.t. $o\mathcal{E}_1$, outputs v with overwhelming probability. Thus, $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ extracts v with overwhelming probability. Moreover, $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ belongs to the class $\mathcal{C}^{\wedge}_{S_2,S_2}$, since $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{S_2,S_2}$ and the rest of computation by $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ takes $\mathsf{poly}(n)$ time. Hence, the scheme $\langle \widehat{C}, \widehat{R}\rangle$ is (S_2, S_2)-over-extractable.

□

Next, we establish the non-malleability of the scheme $\langle \widehat{C}, \widehat{R} \rangle$.

Theorem 10. *$\langle \widehat{C}, \widehat{R} \rangle$ is concurrent $\mathcal{C}^{\wedge}_{d_4,d_4}$-non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$.*

Theorem 11. *$\langle \widehat{C}, \widehat{R} \rangle$ is concurrent $\mathcal{C}^{\wedge}_{d_4,d_4}$-non-malleable (w.r.t. commitment).*

In order to prove concurrent non-malleability w.r.t. commitment, Lin, Pass and Venkitasubramaniam [LPV08] showed that it is sufficient to prove non-malleability against adversaries participating in one left interaction and many right interactions. We refer to such an adversary as a *one-many* adversary. More precisely, they presented a reduction that, given an adversary A and a distinguisher D that break concurrent non-malleability, builds a one-many adversary $\widetilde{A}$ and a distinguisher $\widetilde{D}$ that violate one-many non-malleability. Their reduction blows up the size and the depth of the adversary $\widetilde{A}$ and the distinguisher $\widetilde{D}$ (over A and D respectively) by a $\mathsf{poly}(n)$ factor and thereby incurs a polynomial loss in security. We claim that the same reduction applies to the new notion of non-malleability w.r.t. extraction, therefore establishing that one-many non-malleability w.r.t. extraction implies concurrent non-malleability w.r.t. extraction. Moreover, we consider non-malleability (w.r.t. commitment and extraction) against circuit classes $\mathcal{C}$ which are closed under composition with $\mathcal{P}/\mathrm{poly}$, hence their reduction preserves security in terms of the circuit class against which (concurrent and one-many) non-malleability is considered — a $\mathcal{C}$-one-many non-malleable commitment scheme is $\mathcal{C}$-concurrent non-malleable. We omit a formal proof here but for completeness state the extended version of their theorem below.

Theorem 12 (one-many to concurrent [LPV08])**.** *Let $\langle \widehat{C}, \widehat{R} \rangle$ be a commitment scheme and $\mathcal{C}$ be a class of circuits that is closed under composition with $\mathcal{P}/\mathrm{poly}$.*

1. *If $\langle \widehat{C}, \widehat{R} \rangle$ is $\mathcal{C}$-one-many non-malleable then it is also $\mathcal{C}$-concurrent non-malleable.*
2. *If $\langle \widehat{C}, \widehat{R} \rangle$ is $\mathcal{C}$-one-many non-malleable w.r.t. extraction (by an extractor $\widehat{o\mathcal{E}}_{\mathsf{NM}}$) then it is also $\mathcal{C}$-concurrent non-malleable w.r.t. extraction (by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$).*

Proof of Theorem 10,11: Let us consider a fixed family of circuits $A = \{A_n\}_{n\in\mathbb{N}}$ belonging to the class $\mathcal{C}^{\wedge}_{d_4,d_4}$ which participates in one left interaction and $m = \mathsf{poly}(n)$ right interactions, and any fixed sequences of values $\{v_0\}_{n\in\mathbb{N}}$ and $\{v_1\}_{n\in\mathbb{N}}$. By Theorem 12, to show Theorems 10, 11, it suffices to prove the the following indistinguishabilty:

$$\left\{\mathsf{emim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_0)\right\}_n \approx_c \left\{\mathsf{emim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_1)\right\}_n \tag{9}$$

$$\left\{\mathsf{mim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_0)\right\}_n \approx_c \left\{\mathsf{mim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_1)\right\}_n \tag{10}$$

We prove the above indistinguishability via a sequence of hybrids $\{H_j(v)\}_{j\in[7]}$ for $v \in \{v_0, v_1\}$, where $H_0(v)$ is identical to an honest man-in-the-middle execution with A where it receives a commitment to v in the left interaction, and $H_j(v)$ for each $1 \le j \le 6$ runs a man-in-the-middle execution with A where the left interaction is gradually simulated. For notational convenience, we use x to denote a random variable in the left interaction, and $\tilde{x}_i$ the corresponding random variable in the i'th right interaction. For instance, h denote the hash function sent by A in the left interaction, while $\tilde{h}_i$ denotes that sent by the honest receiver in the i'th right interaction. Moreover, for each hybrid $H_j(v)$, we denote by $\mathsf{mim}^{A}_{H_j}(v)$ (and respectively, $\mathsf{emim}^{A}_{H_j}(v)$) the random variables

that describe the view of A and the values $\{\widetilde{v}_i\}_{i\in[m]}$ committed to in (or respectively$\{\widetilde{v}'_i\}_{i\in[m]}$ extracted from) the right interactions. Again, for every right interaction i, if the interaction is not accepting or its identity $\tilde{\mathsf{id}}_i$ equals to the left identity id, then $\widetilde{v}'_i = \widetilde{v}_i = \perp$; we say that a right intearction is *successful* if this case does not happen.

To show indistinguishability as described in Equation 10 and 9, we prove in Lemma 2 that the view of A and the values extracted from right interactions are indistinguishable in neighboring hybrids $H_j(v)$ and $H_{j+1}(v)$ for the same v, and statistically close in $H_6(v_1)$ and $H_5(v_0)$ — this establishes Equation 9. Furthermore, we show that in every hybrid $H_j(v)$, values extracted from right interactions are actually identical to the actual values committed in right interactions, except with negligible probability. This shows that the emim and mim random variables are statistically close (as stated in Lemma 3) and hence establishes Equation 10.

Lemma 2. *For $v \in \{v_0, v_1\}$ and $j \in [6]$, the following are computationally indistinguishable,*

$$\mathsf{emim}^A_{H_j}(v) \; ; \; \mathsf{emim}^A_{H_{j+1}}(v) \; ,$$

and $\mathsf{emim}^A_{H_0}(v) = \mathsf{emim}^A_{\langle \widehat{C}, \widehat{R}\rangle}(v)$ and $\mathsf{emim}^A_{H_6}(v) = \mathsf{emim}^A_{H_5}(v_0)$.

Lemma 3. *For $v \in \{v_0, v_1\}$ and $j \in [7]$, the following are statistically close,*

$$\mathsf{emim}^A_{H_j}(v) \; ; \; \mathsf{mim}^A_{H_j}(v).$$

Towards proving the above two lemmas, we will maintain a *soundness invariant* throughout all hybrids. Recall that the protocol requires a committer to prove using ZAP that one of the following two statements is true; we refer to the first the honest statement and the second the fake statement.

The honest statement: either it has committed to v in $c1$ (of ECom_1) and to a decomitment $(v, d1)$ of $c1$ in $c3$ (of ECom_3),

The fake statement: or it has committed to a collision $s = (x_1, x_2)$ of the hash function h in $c2$ (of ECom_2), to a decommitment $(s, d2)$ of $c2$ in $c4$ (of ECom_4), and to a decommitment $((s, d2), d4)$ of $c4$ in $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ (of $\langle C, R, \rangle$).

No-fake-witness Invariant. We say that A commits to a fake witness in a right interaction i, if the value committed to by A in the non-malleable commitment $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a decommitment $((\tilde{s}_i, \tilde{d2}_i), \tilde{d4}_i)$ of $\tilde{c4}_i$ satisfying that $\widetilde{s}_i$ is a collision of $\widetilde{h}_i$, $(\widetilde{s}_i, \widetilde{d2}_i)$ is a decommitment of $\widetilde{c2}_i$, and $((\widetilde{s}_i, \widetilde{d2}_i), \widetilde{d4}_i)$ is a decommitment of $\widetilde{c4}_i$.

Invariant 1 (No-fake-witness invariant)**.** *In $H_j(v)$, the probability that there exists a right interaction i that is successful and A commits to a fake witness in it is negligible.*

We show below that this invariant holds in all hybrids. The reason that we maintain Invariant 1 is that it enforces the man-in-the-middle attacker to always prove the honest statement in every successufl right interaction. When this is the case, we show that the values extracted from the right interactions are identical to the values committed to in the right interactions except from negligible probability. Formally,

Claim 1. *In every hybrid $H_j(v)$, if Invaraint 1 holds, then $\mathsf{emim}_{H_j}(v)$ and $\mathsf{mim}_{H_j}(v)$ are statistically close.*

Proof. It suffices to argue that in $H_j(v)$, in every right interaction i, the values $\widetilde{v}'_i$ extracted from this right interaction is identical to the value $\widetilde{v}_i$ committed in this right interaction, except with negligible probability. Note that if a right interaction i is not successful, then $\widetilde{v}'_i = \widetilde{v}_i = \bot$. Otherwise, if a right interaction i is successful, by the definition of the extractor $\widehat{o\mathcal{E}_{\mathsf{NM}}}$ of $\langle \widehat{C}, \widehat{R} \rangle$, the extracted value $\widetilde{v}'_i$ is the value extracted by $o\mathcal{E}_1$ from the ECom_1 commitment $\widetilde{c1}_i$. Under Invariant 1, A does not commit to a fake witness in this (successful) interaction (and hence the fake statement is false). Thus, by the soundness of ZAP, the honest statement must hold that $\widetilde{c3}_i$ commits to a valid decommitment of $\widetilde{c1}_i$ which implies that $\widetilde{c1}_i$ is a valid commitment. In this case, by the over-extractability of ECom_1 w.r.t. $o\mathcal{E}_1$, the value $\widetilde{v}'_i$ extracted from $\widetilde{c1}_i$ is exactly the committed value $\widetilde{v}_i$ except with negligible probability. That is, $\widetilde{v}'_i$ and $\widetilde{v}_i$ are identical except with negligible probability.

Therefore, under Invariant 1, the random variable $\mathsf{emim}^A_{H_j}(v)$ is identical to $\mathsf{mim}^A_{H_j}(v)$, except with negligible probability. □

Thus, showing Lemma 3 boils down to establishing Invariant 1. Towards this goal we further observe that Invariant 1 follows from the following invariant which will be easier to prove. Instead of reasoning about A committing to a fake witness, we keep the invariant that the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is NOT a fake witness.

Invariant 2. *In $H_j(v)$, the probability that there exists a right interaction i that is successful and the value* extracted *from the non-malleable commitment $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ in this session is a fake witness is negligible.*

Claim 2. *In every hybrid $H_j(v)$, if Invaraint 2 holds, then Invaraint 1 also holds.*

Proof. For every right interaction k, consider two cases:

- If the non-malleable commitment $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$ in this right interaction is valid, by the over-extractability property of $\langle C, R \rangle$ w.r.t. extractor $o\mathcal{E}_{\mathsf{NM}}$ the value extracted from it is exactly equal to the value committed, except with negligible probability. Therefore, if the value *extracted* is not a fake witness, neither is the value *committed*, except with negligible probability.
- If the non-malleable commitment $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$ is not valid, the value committed is $\bot$ and cannot be a fake witness.

Hence, Invariant 2 implies Invariant 1. □

Combining the above two claims, we have,

Lemma 4. *For $v \in \{v_0, v_1\}$ and $j \in [7]$, if Invariant 2 holds in hybrid $H_j(v)$ then $\mathsf{emim}^A_{H_j}(v)$ and $\mathsf{mim}^A_{H_j}(v)$ are statistically close.*

Therefore, to show Theorem 10 and Theorem 11, it boils down to prove Lemma 2 and that Invariant 2 holds in all hyrbids. Next, we describe our hybrids $\{H_j(v)\}_{j\in[7]}$ and show that Lemma 2 and Invariant 2 indeed hold.

Hybrid $H_0(v)$: Hybrid $H_0(v)$ emulates an honest MIM execution with A by honestly committing the value v on the left and simulating honest receivers on the right. Therefore,

$$\mathsf{emim}^A_{H_0}(v) = \mathsf{emim}^A_{\langle \widehat{C}, \widehat{R} \rangle}(v) \ .$$

Next, we show that Invariant 2 holds in $H_0(v)$. Infact we show that the value extracted from the ECom_2 commitment $\widetilde{c2}_k$ in any right interaction k is not a collision of the hash function $\widetilde{h}_k$, which implies Invariant 2. At a high level this readily follows from the fact that the collision-resistance hash function is more secure than ECom_2, $h \succ \mathsf{ECom}_2$ (see Figure 2 (iii)). This is because if in some right intearction k, the attack commits to a collision of $\widetilde{h}_k$ using ECom_2, then we can construct a non-uniform circuit that violates the collision-resistance of $\widetilde{h}_k$ by extracting from $\widetilde{c2}_k$.

Claim 3. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_0(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. We show that in $H_0(v)$ the probability that there exists a right interaction k that is successful and the value extracted from $\widetilde{c2}_k$ is a collision of the hash function $\widetilde{h}_k$ in this right interaction — refer to this event bad — is negligible. Then, the claim follows, since whenever the value extracted from the non-malleable commitment in a successful right interaction k is a fake witness, event bad must occur.

Now suppose for contradiction that there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ such that bad occurs with probability $1/p(n)$ in $H_0(v)$. Then, using A, we construct a non-uniform circuit $B = \{B_n\}_{n\in\mathbb{N}} \in \mathcal{C}_{S_{\mathsf{CRH}}}$ that outputs a collision for a hash function sampled from honestly from $\mathcal{H}$ (using D_n) with probability at least $1/p(n)$. More concretely, B with v and k hard-wired in it, on receiving an honestly sampled hash function h^*, emulates $H_0(v)$ for A except for the kth right interaction. In the kth right interaction, B honestly computes the first message $\widetilde{a}_{\mathsf{NM}k}$ of $\langle C, R\rangle$ and the first message $\widetilde{a}_{\mathsf{ZAP}k}$ of ZAP (as in $H_0(v)$) and sends the tuple $(\widetilde{h}_k = h^*, \widetilde{a}_{\mathsf{ZAP}k}, \widetilde{a}_{\mathsf{NM}k})$ as its first round message to A. On receiving the second round message from A in the kth interaction, B runs the extractor $o\mathcal{E}_2$ on $\widetilde{c2}_k$ and returns the extracted value as its output (irrespective of whether the right interaction k is successful or not). Note that B perfectly emulates $H_0(v)$ for A as the distribution of hash function received by B is identical to the distribution of the hash function sent by the honest receiver $\widehat{R}$ of $\langle \widehat{C}, \widehat{R}\rangle$. Then by our hypothesis, the extracted value is a collision of the function $\widetilde{h}_k = h^*$ with probability at least $1/p(n)$.

Furthermore, we argue that B belongs to the circuit class $\mathcal{C}_{S_{\mathsf{CRH}}}$: B internally runs A and $o\mathcal{E}_2$, and the rest of computation performed by B for emulating $H_0(v)$ takes $\mathsf{poly}(n)$ time. Since $o\mathcal{E}_2 \in \mathcal{C}^{\wedge}_{d_1, S_{\mathsf{CRH}}}$ and $A \in \mathcal{C}^{\wedge}_{d_4, S_4}$ we have,

$$\begin{aligned}
\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(o\mathcal{E}_2) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_4) + \mathsf{poly}(S_{\mathsf{CRH}}) \\
&< \mathsf{poly}(S_{\mathsf{CRH}}) \qquad (\text{since, } S_{\mathsf{CRH}} >> d_4 \text{ from Equation (8)})
\end{aligned}$$

Therefore, B belongs to the class $\mathcal{C}_{S_{\mathsf{CRH}}}$ which contradicts the collision-resistance of $\mathcal{H}$. □

Hybrid $H_1(v)$: Hybrid $H_1(v)$ proceeds identically to $H_0(v)$ except that the ECom_2 commitment $c2$ sent to A in the left interaction is generated differently. In $H_0(v)$, $c2$ is a commitment to a random string $r1$ whereas in $H_1(v)$ $c2$ is a commitment to a collision s of the hash function h (received as non-uniform advice). The rest of the execution is simulated identically to $H_0(v)$. We note that only difference between hybrids $H_0(v)$ and $H_1(v)$ is the commitment $c2$ which in $H_0(v)$ commits to a random string $r1$ and in $H_1(v)$ commits to a collision s of the hash function h.

First, we show that Invariant 2 holds in $H_1(v)$. Infact we show that the value extracted from the ECom_4 commitment $\widetilde{c4}_k$ in any right interaction k is not a decommitment of $\widetilde{c2}_k$ to a collision of the hash function $\widetilde{h}_k$, which implies Invariant 2. At a high level this follows from the fact that ECom_2 is more secure than ECom_4, $\mathsf{ECom}_2 \succ \mathsf{ECom}_4$ (see Figure 2 (iii)), and the trick that the reduction can receive a collision of h as a non-uniform advice. Suppose not that in $H_1(v)$, the value extracted from $\widetilde{c4}_k$ in some right interaction k satisfy the condition above with $1/\mathsf{poly}(n)$ probability. By Claim 3, this happens with only negligible probability in $H_0(v)$. Then we can construct a non-uniform circuit that violates the hiding of ECom_2 by extracting from $\widetilde{c4}_k$.

Claim 4. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_1(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. We show that in $H_1(v)$ the probability that there exists a right interaction k that is successful and the value extracted from $\tilde{c4}_k$ is a decommitment of $\tilde{c2}_k$ to a collision of the hash function $\tilde{h}_k$ in this right interaction — refer to this event bad — is negligible. Then, the claim follows, since whenever the value extracted from the non-malleable commitment in a successful right interaction k is a fake witness, event bad must occur. Towards showing this, first observe that by Claim 3 and the same argument, in $H_0(v)$, the probability that bad occurs is negligible.

Now suppose for contradiction that there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ such that bad occurs with probability $1/p(n)$ in $H_1(v)$. Consider the set Γ of prefixes of transcripts up to the point where the first message in the left interaction is sent. By a standard averaging argument, there must exist a $1/2p(n)$ fraction of prefixes ρ in Γ, such that, conditioned on ρ occurring in $H_1(v)$, the probability that bad occurs is at least $1/2p(n)$. Therefore, there exist at least a $1/3p(n)$ fraction of prefixes ρ in Γ, such that, conditioned on ρ occurring in both $H_0(v)$ and $H_1(v)$, the probability that bad occurs jumps from negligible to $1/2p(n)$. Fix one such prefix ρ; let h be the hash function contained in the first message in the left interaction in ρ and $s = (x_1, x_2)$ be a collision of h. Then, using A, the prefix ρ and its collision s, we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_2, S_2}$ that violates the hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ with advantage at least $1/3p(n)$.

The circuit B with v, k, ρ, and s hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ and internally emulates an execution of $H_1(v)$ with A as follows: [9]

- Step 1: Feed A with messages in ρ; let $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ be the left first message.
- Step 2: It samples a random string $r1$, and in the hiding game of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ it sends $r1$ and $s = (x_1, x_2)$ as challenges and receives a commitment c^* to either $r1$ or s.
- Step 3: B generates the second message of the left interaction identically to $H_1(v)$ except that it embeds c^* as the ECom_2 commitment in the message. That is, B computes $(c1, c3, c4, b_{\mathsf{NM}})$ as in $H_1(v)$ (and $H_0(v)$) and then computes the second message of ZAP (b_{ZAP}) by setting $c2 = c^*$. It then sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as second round message in the left interaction to A.
- Step 4: Once, B receives the second round message in the kth right interaction, if the interaction is not successful then B outputs a random bit. Otherwise, it runs the extractor $o\mathcal{E}_4$ on $\tilde{c4}_k$ and outputs 1 iff the extracted value is a decommitment of $\tilde{c2}_k$ to a collision of the hash function $\tilde{h}_k$ in right interaction k.

[9] For right interactions, B sends the first-round message by running the honest receiver $\widehat{R}$.

It is easy to see that if B receives a commitment to the random string $r1$, then it perfectly emulates $H_0(v)$ conditioned on ρ occurring for A and if it receives a commitment to the solution s which is a collision of h then it perfectly emulates $H_1(v)$ conditioned on ρ occurring for A. As argued before, the probability that bad occurs jumps from negligible to $1/2p(n)$. Therefore, B has advantage at least $1/3p(n)$ in violating the hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$.

Moreover, we show that $B \in \mathcal{C}^{\vee}_{d_2,S_2}$: B internally runs $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$, $o\mathcal{E}_4 \in \mathcal{C}^{\wedge}_{d_3,S'_4}$, and the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}\mathsf{dep}(B) &\leq \mathsf{dep}(A) + \mathsf{dep}(o\mathcal{E}_4) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_4) + \mathsf{poly}(d_3)\\ &< \mathsf{poly}(d_2) \qquad (\text{since, } d_2 >> d_4, d_3 \text{ from Equation (8)})\end{aligned}$$

and $\mathsf{size}(B) = \mathsf{poly}(S'_4) < \mathsf{poly}(S^*)$. Therefore, B belongs to the circuit class $\mathcal{C}_{d_2}$ (resp., $B \in \mathcal{C}^{\vee}_{d_2,S_2}$) which contradicts the $\mathcal{C}^{\vee}_{d_2,S_2}$-hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$. Hence, the claim holds. □

Next we show that $\mathsf{emim}^A_{H_0}(v)$ and $\mathsf{emim}^A_{H_1}(v)$ are indistinguishable, that is, view of A and the values extracted from ECom_1 commitments in every successful right interactions is indistinguishable in $H_0(v)$ and $H_1(v)$. This essentially follows from the same proof as Claim 4, but now relying on the fact that ECom_2 is more secure than ECom_1, $\mathsf{ECom}_2 \succ \mathsf{ECom}_1$ (see Figure 2 (iii)),

Claim 5. *For $v \in \{v_0, v_1\}$, the following are indistinguishable,*

$$\mathsf{emim}^A_{H_0}(v); \mathsf{emim}^A_{H_1}(v) \ .$$

Proof. Let us assume for contradiction that there exists a polynomial p and a distinguisher $D \in \mathcal{P}/\text{poly}$ such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ such that D distinguishes $\mathsf{emim}^A_{H_0}(v)$ from $\mathsf{emim}^A_{H_1}(v)$ with probability $\frac{1}{p(n)}$.

Now, consider the set Γ of prefixes of transcripts up to the point where the first message in the left interaction is sent. By a standard averaging argument, there must exist a $1/2p(n)$ fraction of prefixes ρ in Γ, such that, conditioned on ρ occurring in both $H_0(v)$ and $H_1(v)$, the probability that D distinguishes the distributions is at least $1/2p(n)$. Fix one such prefix ρ; let h be the hash function contained in the first message in the left interaction in ρ and $s = (x_1, x_2)$ be a collision of h. Then, using A, the prefix ρ and its collision s, we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_2,S_2}$ that violates the hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ with advantage at least $1/3p(n)$.

The circuit B is similar in spirit to the circuit described in the proof of Claim 4. B with v, k, ρ, and s hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ and internally emulates an execution of $H_1(v)$ with A as follows:

- Steps 1,2 and 3 are identical to the hiding circuit described in Claim 4.
- Step 4: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\widetilde{c1}_i$ to obtain values $\widetilde{v}'_i$. For every unsuccessful right interaction i, B sets $\widetilde{v}'_i = \perp$.
- Step 5: B then runs D with the view of A and the values $\{\widetilde{v_i}'\}_{i\in[m]}$ as inputs, and returns the output of D as its output.

It is easy to see that if B receives a commitment to the random string $r1$, then it perfectly emulates $H_0(v)$ conditioned on ρ occurring for A and if it receives a commitment to the solution s which is a collision of h then it perfectly emulates $H_1(v)$ conditioned on ρ occurring for A. Moreover, for every successful interaction i, B sets $\widetilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$ and for every unsuccessful interaction, it sets $\widetilde{v}'_i = \perp$. Therefore, the input to D (by B) is identical to $\mathsf{emim}^A_{H_0}(v)$ in the former case and it is identical to $\mathsf{emim}^A_{H_1}(v)$ in the latter case. Since D distinguishes the distributions with probability $1/2p(n)$, B wins the hiding game with advantage at least $1/3p(n)$.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_2,S_2}$: Apart from running A, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\widetilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{S_2,S_2}$, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_4) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_2)\\ &< \mathsf{poly}(S_2) \qquad (\text{since, } S_2 >> d_4 \text{ from Equation (8)})\end{aligned}$$

Therefore, B belongs to the circuit class C_{S_2} (resp., $B \in \mathcal{C}^{\vee}_{d_2,S_2}$) which contradicts the $\mathcal{C}^{\vee}_{d_2,S_2}$-hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$. Hence, the claim holds. □

Hybrid $H_2(v)$**:** Hybrid $H_2(v)$ proceeds identically to $H_1(v)$ except that the ECom_4 commitment $c4$ sent to A in the left interaction is generated differently. In $H_1(v)$, $c4$ is a commitment to a random string $r3$ whereas in $H_2(v)$ $c4$ is a commitment to a decommitment of $c2$ to a collision s of the hash function h. More precisely, $H_2(v)$ first finds a collision s for the function h and then commits to s using ECom_2 under $c2$. Then it commits to the decommitment of $c2$ under $c4$. The rest of the execution is simulated identically to $H_1(v)$. We note that only difference between hybrids $H_1(v)$ and $H_2(v)$ is the commitment $c4$ which in $H_1(v)$ commits to a random string $r3$ and in $H_2(v)$ commits to a decommitment of $c2$ to a collision s of h.

First, we show that Invariant 2 holds in $H_2(v)$. At a high level this follows from the fact that ECom_4 is more secure than $\langle C, R\rangle$, $\mathsf{ECom}_4 \succ \langle C, R\rangle$ (see Figure 2 (iii)). Suppose that Invariant 2 does not hold in $H_2(v)$. This means that the value extracted from the non-malleable commitment in some right session k is a fake witness with probability $1/\mathsf{poly}(n)$ in $H_2(v)$, but negligible in $H_1(v)$ by Claim 4. Then, we can construct a non-uniform circuit B that violates the hiding of ECom_4 by extracting from the non-malleable commitment. One sight difference from the proof of Claim 4 is that since ECom_4 is also more secure than h, $\mathsf{ECom}_4 \succ h$ (see Figure 2 (iii)), the reduction B can afford to find collision of h internally, instead of receiving it as a non-uniform advice.

Claim 6. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_2(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ and a right interaction k such that k is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$, is a fake witness with probability at least $1/p(n)$. Then, using A we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_4,S_4}$ that violates the hiding of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$ with advantage at least $1/2p(n)$.

The circuit B with v and k hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$ and internally emulates an execution of $H_2(v)$ with A as follows:

- Step 1: On receiving the first message $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a collision s for the hash function h via brute-force.
- Step 2: It computes commitment $c2$ to the collision s. Let $d2$ be the corresponding decommitment string.
- Step 3: It samples a random string $r3$, and in the hiding game of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$ it sends $r3$ and $(s, d2)$ (decommitment of $c2$ to s) as challenges and receives a commitment c^* to either $r3$ or s.
- Step 4: B generates the second message of the left interaction identically to $H_2(v)$ except that it embeds c^* as the ECom_4 commitment in the message. That is, B computes $(c1, c3, b_{\mathsf{NM}})$ as in $H_2(v)$ (and $H_1(v)$) and then computes the second message of ZAP (b_{ZAP}) by setting $c4 = c^*$. It then sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as second round message in the left interaction to A.
- Step 5: Once, B receives the second round message in the kth right interaction, if the interaction is not successful then B outputs a random bit. Otherwise, it runs the extractor $o\mathcal{E}_{\mathsf{NM}}$ on $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$ and outputs 1 iff the extracted value is a fake witness (i.e., B outputs 1 iff the extracted value is a decommitment of $\widetilde{c4}_k$ to a decommitment of $\widetilde{c2}_k$ to a collision $\widetilde{s}_k$ of $\widetilde{h}_k$.

It is easy to see that if B receives a commitment to the random string $r3$, then it perfectly emulates $H_1(v)$ for A and if it receives a commitment to the decommitment of $c2$ to a collision s of h then it perfectly emulates $H_2(v)$ for A. By Claim 4, in the former case, the extracted value is a fake witness with only negligible probability. Therefore, B outputs 1 with negligible probability. In the latter case, by our assumption that the right interaction k is successful and the value extracted is a fake witness with probability $1/p(n)$; B outputs 1 with probability at least $1/p(n)$. Therefore, B has advantage at least $1/2p(n)$ in violating the hiding of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$.

Moreover, we show that $B \in \mathcal{C}^{\vee}_{d_4,S_4}$: B internally runs $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$, $o\mathcal{E}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{S'_{\mathsf{NM}},S'_{\mathsf{NM}}}$, finds a collision for h using a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$ the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(o\mathcal{E}_{\mathsf{NM}}) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_4) + \mathsf{poly}(S'_{\mathsf{NM}}) + \mathsf{poly}(S'_{\mathsf{CRH}})\\ &< \mathsf{poly}(S_4) \qquad (\text{since, } S_4 >> S'_{\mathsf{NM}}, S'_{\mathsf{CRH}}, d_4 \text{ from Equation (8)})\end{aligned}$$

Therefore, B belongs to the circuit class $\mathcal{C}_{S_4}$ (resp., $B \in \mathcal{C}^{\vee}_{d_4,S_4}$) which contradicts the $\mathcal{C}^{\vee}_{d_4,S_4}$-hiding of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$. Hence, the claim holds. □

Next we show that $\mathsf{emim}^A_{H_1}(v)$ and $\mathsf{emim}^A_{H_2}(v)$ are indistinguishable, that is, view of A and the values extracted from ECom_1 commitments in every successful right interactions is indistinguishable in $H_1(v)$ and $H_2(v)$. The proof is essentially the same as that for Claim 6, except it now relies on the fact that $\mathsf{ECom}_4 \succ \mathsf{ECom}_1$ (and $\mathsf{ECom}_4 \succ h$; see Figure 2 (iii)).

Claim 7. *For $v \in \{v_0, v_1\}$, the following are indistinguishable,*

$$\mathsf{emim}^A_{H_1}(v); \mathsf{emim}^A_{H_2}(v) \ .$$

Proof. Let us assume for contradiction that there exists a polynomial p and a distinguisher $D \in \mathcal{P}/\text{poly}$ such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ such that D distinguishes $\mathsf{emim}^A_{H_1}(v)$ from $\mathsf{emim}^A_{H_2}(v)$ with probability $\frac{1}{p(n)}$. Then using A and D, we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_4,S_4}$ that violates the hiding of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$ with non-negligible advantage $\frac{1}{p(n)}$. B is similar in spirit to the circuit described in the proof of Claim 6.

B with v and k hard-wired in it, participates in the hiding game of ECom_4 and internally emulates an execution of $H_2(v)$ with A as follows:

- Steps 1,2, 3 and 4 are identical to the hiding circuit described in Claim 4.
- Step 5: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\widetilde{c1}_i$ to obtain values $\widetilde{v}'_i$. For every unsuccessful right interaction i, B sets $\widetilde{v}'_i = \perp$.
- Step 6: B then runs D with the view of A and the values $\{\widetilde{v_i}'\}_{i\in[m]}$ as inputs, and returns the output of D as its output.

It is easy to see that if B receives a commitment to the random string $r3$, then it perfectly emulates $H_1(v)$ for A and if it receives a commitment to the decommitment of $c2$ to a collision s of h then it perfectly emulates $H_2(v)$ for A. Moreover, for every successful interaction i, B sets $\widetilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$ and for every unsuccessful interaction, it sets $\widetilde{v}'_i = \perp$. Therefore, the input to D (by B) is identical to $\mathsf{emim}^A_{H_1}(v)$ in the former case and it is identical to $\mathsf{emim}^A_{H_2}(v)$ in the latter case. Since D distinguishes the distributions with probability $1/p(n)$, B wins the hiding game with advantage at least $1/p(n)$.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_4,S_4}$: Apart from running A and finding a collision for h, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\widetilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$, $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{S_2,S_2}$ and a collision for h can be found by a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$, we have,

$$\begin{aligned}
\mathsf{size}(B) &= \mathsf{size}(A) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_4) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_2) + \mathsf{poly}(S'_{\mathsf{CRH}}) \\
&< \mathsf{poly}(S_4) \qquad (\text{since, } S_4 >> S'_{\mathsf{NM}}, S'_{\mathsf{CRH}}, d_4 \text{ from Equation (8)})
\end{aligned}$$

Therefore, B belongs to the circuit class C_{S_4} (resp., $B \in \mathcal{C}^{\vee}_{d_4,S_4}$) which contradicts the $\mathcal{C}^{\vee}_{d_4,S_4}$-hiding of $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$. Hence, the claim holds. □

Hybrid $H_3(v)$: Hybrid $H_3(v)$ proceeds identically to $H_2(v)$ except that the second message b_{NM} of $\langle C, R\rangle$ sent to A in the left interaction is generated differently. In $H_2(v)$, b_{NM} is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to a random string $r2$ whereas in $H_3(v)$ b_{NM} is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commit to a decommitment of $c4$ to a decommitment of $c2$ to a collision s of the hash function h. More precisely, $H_3(v)$ generates a commitment $c2$ to the collision s (obtained by brute-force search). Let $d2$ be the corresponding decommitment string. Then, $H_3(v)$ computes the commitment $c4$ to the decommitment $(s, d2)$ of $c2$. Let $d4$ be the corresponding decommitment string. Then, given a_{NM}, $H_3(v)$ computes the second message b_{NM} to commit to $((s, d2), d4)$. The rest of the execution is simulated identically to $H_2(v)$. We note that only difference between hybrids $H_2(v)$ and $H_3(v)$ is the second message b_{NM} which is such that in $H_1(v)$ $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to a random string $r2$ whereas in $H_2(v)$ $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to $((s, d2), d4)$.

First, we show that Invariant 2 holds in $H_3(v)$. At a high-level, this follows from the one-one non-malleability w.r.t. extraction of $\langle C, R\rangle$. Suppose that Invariant 2 does not hold in $H_3(v)$ then there exists a right interaction k such that the probability that it is successful and the value extracted the non-malleable commitment contained in this session is a fake witness is $1/\mathsf{poly}(n)$ in $H_3(v)$ and is negligible in $H_2(v)$ (by Claim 6). This violates the one-one non-malleability w.r.t. extraction of $\langle C, R\rangle$ as we formally show below.

Claim 8. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_3(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ and a right interaction k such that k is successful and the value $((\widetilde{s}'_k, \widetilde{d2}'_k), \widetilde{d4}'_k)$, extracted from $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$, is a fake witness with probability at least $1/p(n)$. Then, using A we construct a non-uniform circuit $A_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{S_{\mathsf{NM}}, S_{\mathsf{NM}}}$, that participates in one left interaction with C and one right interaction with R, and a distinguisher D_{NM} that violate the one-one non-malleability of $\langle C, R\rangle$ with advantage at least $1/2p(n)$. We detail the circuits A_{NM} and D_{NM} below.

The circuit A_{NM} with v and k hard-wired in it, participates in one left interaction with C and one right interaction with R and internally emulates an execution of $H_3(v)$ with A as follows:

- Step 1: A_{NM} waits for A to select identities for the left interaction with C and the kth right interaction with R. Let id and $\widetilde{\mathsf{id}}_k$ be the respective identities.
- Step 2: A_{NM} selects identity $\mathsf{id}_l = \mathsf{id}$ for its left interaction and identity $\mathsf{id}_r = \widetilde{\mathsf{id}}_k$ for its right interaction r. On receiving the first-round message $a_{\mathsf{NM}r}$ from R, A_{NM} samples a hash function $\widetilde{h}_k$ and the first message of ZAP, $\widetilde{a}_{\mathsf{ZAP}k}$. It sends the tuple $(\widetilde{h}_k, \widetilde{a}_{\mathsf{NM}k} = a_{\mathsf{NM}r}, \widetilde{a}_{\mathsf{ZAP}k})$ as the first-round message to A in the kth right interaction.
- Step 3: On receiving the first message $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, A_{NM} obtains a collision s for h via brute-force search.
- Step 4: A_{NM} computes commitments $(c1, c2, c3, c4)$ as in $H_3(v)$. Let $d2$ be the decommitment string of the commitment $c2$, which commits to the collison s. Furthermore, let $d4$ be the decommitment string of $c4$ which commits to a decommitment of $c2$.
- Step 5: A_{NM} samples a random string $r2$ and sends $a_{\mathsf{NM}l} = a_{\mathsf{NM}}$ as the first message to C along with the values $r2$ and $((s, d2), d4)$ as challenges and receives the second message $b_{\mathsf{NM}l}$ such that $(a_{\mathsf{NM}l}, b_{\mathsf{NM}l})$ either commit to $r2$ or $((s, d2), d4)$.
- Step 6: A_{NM} computes the second message of ZAP (b_{ZAP}) by setting $b_{\mathsf{NM}} = b_{\mathsf{NM}l}$. Then, it sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as the second round message to A in the left interaction.
- Step 7: On receiving the second message $(\widetilde{c1}_k, \widetilde{c2}_k, \widetilde{c3}_k, \widetilde{c4}_k, \widetilde{b}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{ZAP}k})$ from A in the kth right interaction, B forwards $b_{\mathsf{NM}r} = \widetilde{b}_{\mathsf{NM}k}$ as the second message to R.

The distinguisher D_{NM} with input the view of A_{NM} and the value v'_r, extracted from $(a_{\mathsf{NM}r}, b_{\mathsf{NM}r})$ by $o\mathcal{E}_{\mathsf{NM}}$, runs as follows:

- D_{NM} reconstructs the entire transcript of the kth right interaction of A_{NM} with A from the view.
- If the ZAP proof $(\widetilde{a}_{\mathsf{ZAP}k}, \widetilde{b}_{\mathsf{ZAP}k})$ in the kth interaction is not accepting then D_{NM} outputs a random bit.

- Otherwise, D_{NM} outputs 1 iff the extracted value v'_r is such that it is a decommitment of $\widetilde{c4}_k$ to a decommitment of $\widetilde{c2}_k$ to a collision of the hash function $\widetilde{h}_k$.

It is easy to see that if A_{NM} receives $b_{\mathsf{NM}l}$ such that $(a_{\mathsf{NM}l}, b_{\mathsf{NM}l})$ commit to a random string $r2$ then it perfectly emulates $H_2(v)$ for A and if $b_{\mathsf{NM}l}$ is such that $(a_{\mathsf{NM}l}, b_{\mathsf{NM}l})$ commit to $((s, d2), d4)$ then it perfectly emulates $H_3(v)$ for A. By Claim 6, in the former case, the extracted value v'_r is a fake witness with only negligible probability. Therefore, D_{NM} outputs 1 with negligible probability. In the latter case, by our assumption that the right interaction k is successful and the value extracted is a fake witness with probability $1/p(n)$; D_{NM} outputs 1 with probability at least $1/p(n)$. Therefore, D_{NM} has advantage at least $1/2p(n)$ in distinguishing the two cases. Therefore, A_{NM} and D_{NM} break the one-one non-malleability w.r.t. extraction of $\langle C, R\rangle$.

Moreover, we argue that $A_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{S_{\mathsf{NM}}, S_{\mathsf{NM}}}$ and $D_{\mathsf{NM}} \in \mathcal{P}/\mathsf{poly}$: Firstly, it is easy to see that $D_{\mathsf{NM}} \in \mathcal{P}/\mathsf{poly}$ as all the computation done by D_{NM} only takes polynomial time.

Next, for A_{NM}: A_{NM} internally runs $A \in \mathcal{C}^{\wedge}_{d_4, d_4}$, finds a collision for h using a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$ and the rest of the computation done by A_{NM} takes $\mathsf{poly}(n)$ time. Therefore, the size $\mathsf{size}(A_{\mathsf{NM}})$ of A_{NM} satisfies the following,

$$\begin{aligned}
\mathsf{size}(A_{\mathsf{NM}}) &= \mathsf{size}(A) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_4) + \mathsf{poly}(S'_{\mathsf{CRH}}) \\
&< \mathsf{poly}(S_{\mathsf{NM}}) \qquad (\text{since, } S_{\mathsf{NM}} >> d_4, S'_{\mathsf{CRH}} \text{ from Equation (8)})
\end{aligned} \tag{11}$$

Therefore, A_{NM} belongs to the circuit class $\mathcal{C}^{\wedge}_{S_{\mathsf{NM}}, S_{\mathsf{NM}}}$ which contradicts the $\mathcal{C}^{\wedge}_{S_{\mathsf{NM}}, S_{\mathsf{NM}}}$-one-one non-malleability w.r.t. extraction of $\langle C, R\rangle$. Hence, the claim holds. □

Remark 6. *Note that in the above reduction to one-one non-malleability w.r.t. extraction, we allow A_{NM} to send the challenge values $r2$ and $((s, d2), d4)$ along with the first message a_{NM}. The committer C is expected to commit to either $r2$ or $((s, d2), d4)$. Note that the challenges $r2$ and $((s, d2), d4)$ could* depend *on the right interaction whereas for the notions of non-malleability used in this work, the value committed on the left is independent of the right interaction and fixed before the MIM execution begins. Therefore, the adversary A_{NM} is stronger than the adversaries considered in the non-malleability definitions. However, this gap can be bridged by one of the following,*

1. *Defining non-malleability w.r.t. adversaries that can adaptively sample the challenge values analogous to choosing the identities. We note that all the commitment schemes defined in this work actually satisfy this stronger notion of non-malleability.*
2. *Adopting the approach taken by [COSV16b] in our context: Instead of committing to the decommitment $((s, d2), d4)$ of $c4$ under the non-malleable commitment $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$, we instead commit to a random share s_0 of the decommitment $((s, d2), d4)$ under $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ and send the other share s_1 in the clear to the receiver. Furthermore, the ZAP now proves that either $c3$ commits to a decommitment of $c1$ or that s_1 xored with the value committed under $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ is a decommitment of $c4$ to a decommitment of $c2$ to a collision s. This allows the challenge messages to be fixed before the execution. We omit details here.*

Next we show that $\mathsf{emim}^A_{H_2}(v)$ and $\mathsf{emim}^A_{H_3}(v)$ are indistinguishable, that is, view of A and the values extracted from ECom_1 commitments in every successful right interactions is indistinguishable in $H_2(v)$ and $H_3(v)$. This follows from the fact that $\langle C, R\rangle$ is more secure than ECom_1, $\langle C, R\rangle \succ \mathsf{ECom}_1$ (see Figure 2 (iii)). Therefore, if the distribution of values extracted from the ECom_1 commtiments in the right interactions are distinguishable in H_2 and H_3, one can construct reduction that violates the hiding of $\langle C, R\rangle$ by extracting from the ECom commitments on the right.

Claim 9. *For $v \in \{v_0, v_1\}$, the following are indistinguishable,*

$$\mathsf{emim}^A_{H_2}(v); \mathsf{emim}^A_{H_3}(v) \ .$$

Proof. Let us assume for contradiction that there exists $v \in \{v_0, v_1\}$, a distinguisher $D \in \mathcal{P}/\mathrm{poly}$ and a polynomial p such that D distinguishes $\mathsf{emim}^A_{H_2}(v)$ from $\mathsf{emim}^A_{H_3}(v)$ with probability $\frac{1}{p(n)}$. Then using A and D, we construct a non-uniform circuit $B \in \mathcal{C}^{\wedge}_{S_{\mathsf{NM}}, S_{\mathsf{NM}}}$ that violates the hiding of $\langle C, R\rangle$ with non-negligible advantage $\frac{1}{p(n)}$. B is similar in spirit to the circuit A_{NM} described in the proof of Claim 8.

B with v and k hard-wired in it, participates in the hiding game of $\langle C, R\rangle$ and internally emulates an execution of $H_3(v)$ with A as follows:

- Step 1: On receiving the first message $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a collision s for the hash function h via brute-force.
- Step 2: B computes commitments $(c1, c2, c3, c4)$ as in $H_3(v)$. Let $d2$ be the decommitment string of the commitment $c2$, which commits to the collision s. Furthermore, let $d4$ be the decommitment string of the commitment $c4$ to the decommitment $c2$.
- Step 3: B samples a random string $r2$ and sends a_{NM} as the first message to C along with the values $r2$ and $((s, d2), d4)$ as challenges and receives the second message b_{NM} such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ either commit to $r2$ or $((s, d2), d4)$.
- Step 4: A_{NM} computes the ZAP proof and sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as the second round message to A in the left interaction.
- Step 5: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\widetilde{c1}_i$ to extract values $\widetilde{v}'_i$. For every unsuccessful right interaction i, B sets $\widetilde{v}'_i = \perp$.
- Step 6: B then runs D with the view of A and the values $\{\widetilde{v_i}'\}_{i\in[m]}$ as inputs, and returns the output of D as its output.

It is easy to see that if second message b_{NM} received by B is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commit to a random string $r2$, then B is perfectly emulating $H_2(v)$ for A and if b_{NM} is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to $((s, d2), d4)$, then it perfectly emulating $H_3(v)$ for A. Moreover, for every successful interaction i, B sets $\widetilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$ and for every unsuccessful interaction B sets $\widetilde{v}'_i = \perp$. Therefore, the input to D (by B) is identical to $\mathsf{emim}^A_{H_2}(v)$ in the former case and it is identical to $\mathsf{emim}^A_{H_3}(v)$ in the latter case. Since D distinguishes the distributions with probability $1/p(n)$, B wins the hiding game with advantage at least $1/p(n)$.

Next, we argue that $B \in \mathcal{C}^{\wedge}_{S_{\mathsf{NM}}, S_{\mathsf{NM}}}$: Apart from running A and using a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$ to find the collision s, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\widetilde{c1}_i$, and the rest of the

computation takes polynomial time (including running D). Since, $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{S_2,S_2}$, the size $\mathsf{size}(A_{\mathsf{NM}})$ of A_{NM} satisfies the following,

$$\begin{aligned}\mathsf{size}(A_{\mathsf{NM}}) &= \mathsf{size}(A) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n) \\ &\leq \mathsf{poly}(d_4) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_2) + \mathsf{poly}(S'_{\mathsf{CRH}}) \\ &< \mathsf{poly}(S_{\mathsf{NM}}) \qquad (\text{since, } S_{\mathsf{NM}} >> d_4, S_2, S'_{\mathsf{CRH}} \text{ from Equation (8)})\end{aligned} \tag{12}$$

Therefore, B belongs to the circuit class $\mathcal{C}^{\wedge}_{S_{\mathsf{NM}},S_{\mathsf{NM}}}$ which contradicts the $\mathcal{C}^{\wedge}_{S_{\mathsf{NM}},S_{\mathsf{NM}}}$-hiding of $\langle C, R\rangle$. Hence, the claim holds. □

Hybrid $H_4(v)$: Hybrid $H_4(v)$ proceeds identically to $H_3(v)$ except that the second message b_{ZAP} of ZAP sent to A in the left interaction is generated differently. In $H_3(v)$, b_{ZAP} is computed by proving that $c3$ commits to a decommitment $(v, d1)$ of $c1$ whereas in $H_4(v)$ b_{ZAP} is computed by proving that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to $((s, d2), d4)$ which is a decommitment of $c4$ to a decommitment $(s, d2)$ of $c2$ to the collision s of the hash function h. We note that only difference between hybrids $H_3(v)$ and $H_4(v)$ is the second message b_{ZAP}, or more precisely the witness used to compute the second message b_{ZAP}. In $H_3(v)$, the witness used is the decommitment of $c3$ to $(v, d1)$ whereas in $H_4(v)$ the witness is a decommitment of $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ to $((s, d2), d4)$.

First, we show that Invariant 2 holds in $H_4(v)$. At a high-level, this follows from the witness indistinguishability of ZAP, which holds against subexponentially-sized attackers. Since $\langle C, R\langle$ can be broken in the time that ZAP is secure against, changing the ZAP proof on the left should not change the distribution of values extracted from the right non-malleable commitments. Therefore by Claim 8 that these values are not fake witnesses in $H_3(v)$, the same holds in $H_4(v)$.

Claim 10. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_4(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists a polynomial p such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ and a right interaction k such that k is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$, is a fake witness with probability at least $1/p(n)$. Then, using A we construct a non-uniform circuit $B \in \mathcal{C}_{S^*}$ that violates the $\mathcal{C}_{S^*}$-WI of ZAP with advantage at least $1/2p(n)$.

The circuit B with v and k hard-wired in it, participates in the WI game of ZAP and internally emulates an execution of $H_4(v)$ with A as follows:

- Step 1: On receiving the first message $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a collision s to the hash function h.
- Step 2: B computes commitments $(c1, c2, c3, c4, b_{\mathsf{NM}})$ (as in $H_4(v)$). Let $d1$ be the decommitment string of the commitment $c1$, which commits to the value v, $d4$ be the decommitment of $c4$ which commits to $(s, d2)$ where $d2$ is the decommitment string of the commitment $c2$, which commits to the collison s. Furthermore, let $d3$ and d be the decommitments of $c3$ and $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$.

- Step 3: B sends a_{ZAP} as the first message in the WI game of ZAP with the statement $x = (h, c1, c2, c3, c4, a_{\mathsf{NM}}, b_{\mathsf{NM}})$ and witnesses $w_0 = (v, d1, d3)$ and $w_1 = (((s, d2), d4), d)$. B receives the second message b_{ZAP} of ZAP that is either computed by using the witness w_0 or w_1.
- Step 4: B sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as the second message to A on the left.
- Step 5: Once, B receives the second round message in the kth right interaction, if the interaction is not successful then B outputs a random bit. Otherwise, it runs the extractor $o\mathcal{E}_{\mathsf{NM}}$ on $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$ and outputs 1 iff the extracted value is a fake witness.

It is easy to see that if the second message b_{ZAP} of ZAP is computed using the witness $w_0 = (v, d1, d3)$ then B perfectly emulates $H_3(v)$ for A and if the second message b_{ZAP} of ZAP is computed using the witness $w_1 = (((s, d2), d4), d)$ then B perfectly emulates $H_4(v)$ for A. By Claim 8, in the former case, the extracted value is a fake witness with only negligible probability. Therefore, B outputs 1 with negligible probability. In the latter case, by our assumption that k is successful and the value extracted is a fake witness with probability $1/p(n)$; B outputs 1 with probability at least $1/p(n)$. Therefore, B has advantage at least $1/2p(n)$ in violating the WI of ZAP.

Moreover, we show that $B \in \mathcal{C}_{S^*}$: B internally runs $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$, $o\mathcal{E}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{S'_{\mathsf{NM}},S'_{\mathsf{NM}}}$, obtains a collision for h by using a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$ and the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}
\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{size}(o\mathcal{E}_{\mathsf{NM}}) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_4) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(S'_{\mathsf{NM}}) \\
&< \mathsf{poly}(S^*) \qquad (\text{since, } S^* >> d_4, S'_{\mathsf{CRH}}, S'_{\mathsf{NM}} \text{ from Equation (8)})
\end{aligned}$$

Therefore, B belongs to the circuit class $\mathcal{C}_{S^*}$ which contradicts the $\mathcal{C}_{S^*}$-witness-indistinguishability of ZAP. Hence, the claim holds. □

Next we show that $\mathsf{emim}^A_{H_3}(v)$ and $\mathsf{emim}^A_{H_4}(v)$ are indistinguishable, that is, view of A and the values extracted from ECom_1 commitments in every successful right interactions is indistinguishable in $H_3(v)$ and $H_4(v)$. This follows from essentially the same proof of Claim 8, except that now we use the fact that ZAP is more secure than ECom_1.

Claim 11. *For $v \in \{v_0, v_1\}$, the following are indistinguishable,*

$$\mathsf{emim}^A_{H_3}(v); \mathsf{emim}^A_{H_4}(v) \ .$$

Proof. Let us assume for contradiction that there exists a polynomial p and a distinguisher D such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ such that D distinguishes $\mathsf{emim}^A_{H_3}(v)$ from $\mathsf{emim}^A_{H_4}(v)$ with probability $\frac{1}{p(n)}$. Then using A and D, we construct a non-uniform circuit $B \in \mathcal{C}_{S^*}$ that violates the $\mathcal{C}_{S^*}$-WI of ZAP with advantage at least $1/p(n)$. B is similar in spirit to the circuit described in the proof of Claim 10.

B with v and k hard-wired in it, participates in the WI game of ZAP and internally emulates an execution of $H_4(v)$ with A as follows:

- Steps 1,2,3 and 4 are identical to the circuit described in Claim 10.
- Step 5: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\widetilde{c1}_i$ to extract values $\widetilde{v}'_i$. For every unsuccessful right interaction i, B sets $\widetilde{v}'_i = \perp$.

- Step 6: B then runs D with the view of A and the values $\{\widetilde{v}_i'\}_{i\in[m]}$ as inputs, and returns the output of D as its output.

It is easy to see that if the second message b_{ZAP} of ZAP is computed using the witness $w_0 = (v, d1, d3)$ then B perfectly emulates $H_3(v)$ for A and if the second message b_{ZAP} of ZAP is computed using the witness $w_1 = (((s, d2), d4), d)$ then B perfectly emulates $H_4(v)$ for A. Moreover, for every successful interaction i, B sets $\widetilde{v}_i'$ to the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$ and for every unsuccessful interaction, it sets $\widetilde{v}_i' = \bot$. Therefore, the input to D (by B) is identical to $\mathsf{emim}^A_{H_3}(v)$ in the former case and it is identical to $\mathsf{emim}^A_{H_4}(v)$ in the latter case. Since D distinguishes the distributions with probability $1/p(n)$, B wins the hiding game with advantage at least $1/p(n)$.

Next, we argue that $B \in \mathcal{C}_{S^*}$: Apart from running A and finding a collision for h, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\widetilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{S_2,S_2}$, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{poly}(S'_{\mathsf{CRH}}) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_4) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_2)\\ &< \mathsf{poly}(S^*) \qquad (\text{since, } S^* >> d_4, S_2, S'_{\mathsf{CRH}} \text{ from Equation (8)})\end{aligned}$$

Therefore, B belongs to the circuit class $\mathcal{C}_{S^*}$ which contradicts the $\mathcal{C}_{S^*}$-WI of ZAP. Hence, the claim holds. □

Hybrid $H_5(v)$: Hybrid $H_5(v)$ proceeds identically to $H_4(v)$ except that the ECom_3 commitment $c3$ sent to A in the left interaction is generated differently. In $H_4(v)$ $c3$ is committing to the decommitment $(v, d1)$ of $c1$ whereas in $H_5(v)$ $c3$ is committing to 0^l where l is the length of the decommitment of $c1$. More precisely, $H_5(v)$ computes $(c1, c2, c4, b_{\mathsf{NM}})$ identical to $H_4(v)$. Then, $H_5(v)$ computes the ECom_3 commitment $c3$ to commit to 0^l. The rest of the execution is simulated identically to $H_4(v)$. We note that only difference between hybrids $H_4(v)$ and $H_5(v)$ is the ECom_3 commitment $c3$ which in $H_4(v)$ commits to the decommitment of $c1$ (to the value v) whereas in $H_5(v)$ $c3$ commits to 0^l.

First, we show that Invariant 2 holds in $H_5(v)$. This follows from the fact that $\mathsf{ECom}_3 \succ \langle C, R\rangle$, (see Figure 2 (iii)). Suppose that Invariant 2 does not hold in $H_5(v)$ but holds in $H_4(v)$ by Claim 10, then there exists a right interaction k such that the probability that it is successful and the value extracted from the non-malleable commitment in it is a fake witness jumps from negligible in $H_4(v)$ to $1/\mathsf{poly}(n)$ in $H_5(v)$. Then, we can construct a reduction that violates the hiding of ECom_3 by extracting from the non-malleable commitment in the kth right interaction.

Claim 12. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_5(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there a polynomial p such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ and a right interaction k such that k is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$, is a fake witness with probability at least $1/p(n)$. Then, using A we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_3,S_3}$ that violates the hiding of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ with advantage at least $1/2p(n)$.

The circuit B with v and k hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ and internally emulates an execution of $H_5(v)$ with A as follows:

- Step 1: On receiving the first message $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a collison s to the hash function h.
- Step 2: It computes $(c1, c2, c4, b_{\mathsf{NM}})$ as in $H_5(v)$. Let $d1$ be the decommitment string of the commitment $c1$.
- Step 3: In the hiding game of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$, B sends $(v, d1)$ and 0^l as challenges and receives a commitment c^* to either $(v, d1)$ or 0^l.
- Step 4: B generates the second message of ZAP (b_{ZAP}) by setting $c3 = c^*$. It then sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as second round message in the left interaction to A.
- Step 5: Once, B receives the second round message in the kth right interaction, if the interaction is not successful then B outputs a random bit. Otherwise, it runs the extractor $o\mathcal{E}_{\mathsf{NM}}$ on $(\widetilde{a}_{\mathsf{NM}k}, \widetilde{b}_{\mathsf{NM}k})$ and outputs 1 iff the extracted value is a fake witness.

It is easy to see that if B receives a commitment to $(v, d1)$, then it perfectly emulates $H_4(v)$ for A and if it receives a commitment to 0^l then it perfectly emulates $H_5(v)$ for A. By Claim 10, in the former case, the extracted value is a fake witness with only negligible probability. Therefore, B outputs 1 with negligible probability. In the latter case, by our assumption that the right interaction k is successful and the value extracted is a fake witness with probability $1/p(n)$; B outputs 1 with probability at least $1/p(n)$. Therefore, B has advantage at least $1/2p(n)$ in violating the hiding of ECom_3.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_3,S_3}$: B internally runs $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$, $o\mathcal{E}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{S'_{\mathsf{NM}},S'_{\mathsf{NM}}}$, obtains a collision for h using a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$ and the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(o\mathcal{E}_{\mathsf{NM}}) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_4) + \mathsf{poly}(S'_{\mathsf{NM}}) + \mathsf{poly}(S'_{\mathsf{CRH}})\\ &< \mathsf{poly}(S_3) \qquad (\text{since, } S_3 >> d_4, S'_{\mathsf{NM}}, S'_{\mathsf{CRH}} \text{ from Equation (8)})\end{aligned}$$

Therefore, B belongs to the circuit class $\mathcal{C}_{S_3}$ (resp., $B \in \mathcal{C}^{\vee}_{d_3,S_3}$) which contradicts the $\mathcal{C}^{\vee}_{d_3,S_3}$-hiding of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$. Hence, the claim holds. □

Next we show that $\mathsf{emim}^A_{H_4}(v)$ and $\mathsf{emim}^A_{H_5}(v)$ are indistinguishable, that is, view of A and the values extracted from ECom_1 commitments in every successful right interactions is indistinguishable in $H_4(v)$ and $H_5(v)$. This follows from the same proof as that of Claim 12, except that now it relies on the fact that $\mathsf{ECom}_3 \succ \mathsf{ECom}_1$.

Claim 13. *For $v \in \{v_0, v_1\}$, the following are indistinguishable,*

$$\mathsf{emim}^A_{H_4}(v); \mathsf{emim}^A_{H_5}(v) \ .$$

Proof. Let us assume for contradiction that there exists a polynomial p and a distinguisher $D \in \mathcal{P}/\text{poly}$ such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$, such that D distinguishes $\mathsf{emim}^A_{H_4}(v)$ from $\mathsf{emim}^A_{H_5}(v)$ with probability $\frac{1}{p(n)}$. Then using A and D, we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_3,S_3}$ that violates the hiding of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ with non-negligible advantage $\frac{1}{p(n)}$. B is similar in spirit to the circuit described in the proof of Claim 10.

B with v and k hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ and internally emulates an execution of $H_5(v)$ with A as follows:

- Steps 1-4 are identical to the hiding circuit described in Claim 12.
- Step 5: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\widetilde{c1}_i$ to extract values $\widetilde{v}'_i$. For every unsuccessful right interaction i, B sets $\widetilde{v}'_i = \perp$.
- Step 6: B then runs D with the view of A and the values $\{\widetilde{v_i}'\}_{i\in[m]}$ as inputs, and returns the output of D as its output.

It is easy to see that if B receives a commitment to $(v, d1)$, then it perfectly emulates $H_4(v)$ for A and if it receives a commitment to 0^l then it perfectly emulates $H_5(v)$ for A. Moreover, B for every successful interaction i, sets $\widetilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$ and for every unsuccessful interaction, it sets $\widetilde{v}'_i = \perp$. Therefore, the input to D (by B) is identical to $\mathsf{emim}^A_{H_4}(v)$ in the former case and it is identical to $\mathsf{emim}^A_{H_5}(v)$ in the latter case. Since D distinguishes the distributions with probability $1/p(n)$, B wins the hiding game with advantage at least $1/p(n)$.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_3,S_3}$: Apart from running A and finding a collision for h using a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\widetilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_4,d_4}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{S_2,S_2}$, we have,

$$\begin{aligned}
\mathsf{size}(B) &= \mathsf{size}(A) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(S'_{\mathsf{CRH}}) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_4) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_2) + \mathsf{poly}(S'_{\mathsf{CRH}}) \\
&< \mathsf{poly}(S_3) \qquad (\text{since, } S_3 >> d_4, S_{\mathsf{NM}}, S'_{\mathsf{CRH}} \text{ from Equation (8)})
\end{aligned}$$

Therefore, B belongs to the circuit class $\mathcal{C}_{S_3}$ (resp., $B \in \mathcal{C}^{\vee}_{d_3,S_3}$) which contradicts the $\mathcal{C}^{\vee}_{d_3,S_3}$-hiding of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$. Hence, the claim holds. □

Hybrid $H_6(v)$: Hybrid $H_6(v)$ proceeds identically to $H_5(v)$ except that the ECom_1 commitment $c1$ sent to A in the left interaction is generated differently. In $H_5(v)$, $c1$ is committing to the value v whereas in $H_6(v)$ $c1$ is committing to the value (fixed) v_0 instead. The rest of the execution is simulated identically to $H_5(v)$. We note that the only difference between hybrids $H_5(v)$ and $H_6(v)$ is the ECom_1 commitment $c1$ which in $H_5(v)$ commits to v but in $H_6(v)$ $c1$ commits to v_0.

First, note that $H_6(v)$ is in fact identical to $H_5(v_0)$. Therefore by Claim 12 that Invariant 2 holds in $H_5(v_0)$, we directly have that it holds also in $H_6(v)$.

Claim 14. *For $v \in \{v_0, v_1\}$ and for every right interaction i in $H_6(v)$, the probability that i is successful and the value extracted from $(\widetilde{a}_{\mathsf{NM}i}, \widetilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Next we show that $\mathsf{emim}^A_{H_5}(v)$ and $\mathsf{emim}^A_{H_6}(v)$ are indistinguishable. This follows from the fact that ECom_1 is more secure than ECom_3, $\mathsf{ECom}_1 \succ \mathsf{ECom}_3$ (see Figure 2 (iii)), and the fact that Invariant 2 holds in both $H_5(v)$ and $H_6(v)$. The latter ensures that in every successful right interaction k, the attacker must prove the honest statement using ZAP that $\widetilde{c3}_k$ is valid committing to a valid decommitment of $\widetilde{c1}_k$ in that right interaction. Therefore, in every successful right interaction k, the value extracted from $\widetilde{c3}_k$ and $\widetilde{c1}_k$ are identical. This implies that if the emim random variables are distinguishable in $H_5(v)$ and $H_6(v)$, the values extracted from the right ECom_3 commitments are also distinguishable. Then, we can construct a reduction that violates the hiding of ECom_1 by extracting from the right ECom_3 commitments.

Claim 15. *For $v \in \{v_0, v_1\}$, the following are indistinguishable,*

$$\mathsf{emim}^A_{H_5}(v); \mathsf{emim}^A_{H_6}(v) \ .$$

Proof. Let us assume for contradiction that there exists a polynomial p and a distinguisher $D \in \mathcal{P}/\text{poly}$ such that for infinitely many $n \in \mathbb{N}$ there exists $v \in \{v_0, v_1\}$ such that D distinguishes $\mathsf{emim}^A_{H_0}(v)$ from $\mathsf{emim}^A_{H_1}(v)$ with probability $\frac{1}{p(n)}$.

Now, consider the set Γ of prefixes of transcripts up to the point where the first message in the left interaction is sent. By a standard averaging argument, there must exist a $1/2p(n)$ fraction of prefixes ρ in Γ, such that, conditioned on ρ occurring in both $H_5(v)$ and $H_6(v)$, the probability that D distinguishes the distributions is at least $1/2p(n)$. Fix one such prefix ρ; let h be the hash function contained in the first message in the left interaction in ρ and $s = (x_1, x_2)$ be a collision of h. Then, using A, the prefix ρ and its collision s, we construct a non-uniform circuit $B \in \mathcal{C}_{d_1}$ that violates the hiding of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ with advantage at least $1/3p(n)$.

B with v, k, ρ, and s hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ and internally emulates an execution of $H_6(v)$ with A as follows:

- Step 1: Feed A with messages in ρ; let $(h, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ be the left first message.
- Step 2: B sends v and v_0 as challenges in the hiding game of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ and receives a commitment c^* to either v or v_0.
- Step 3: B generates the second message of the left interaction identically to $H_6(v)$ except that it embeds c^* as the ECom_1 commitment in the message. That is, B computes $(c2, c3, c4, b_{\mathsf{NM}})$ as in $H_6(v)$ (using the collision s received as non-uniform advice) and then computes the second message of ZAP (b_{ZAP}) by setting $c1 = c^*$. It then sends $(c1, c2, c3, c4, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as second round message in the left interaction to A.
- Step 4: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_3$ on $\widetilde{c3}_i$ to extract values $(\widetilde{v}'_i, \widetilde{d1}'_i)$. For every unsuccessful right interaction i, B sets $\widetilde{v}'_i = \perp$.
- Step 4: B then runs D with the view of A and the values $\{\widetilde{v_i}'\}_{i \in [m]}$ as inputs, and returns the output of D as its output.

It is easy to see that if B receives a commitment to v, then it perfectly emulates $H_5(v)$ conditioned on ρ occurring for A and if it receives a commitment to v_0 then it perfectly emulates $H_6(v)$ conditioned on ρ occurring for A. Moreover, for every successful interaction i, B sets $\widetilde{v}'_i$ to the value extracted by $o\mathcal{E}_3$ from $\widetilde{c3}_i$ and for every unsuccessful interaction, it sets $\widetilde{v}'_i = \perp$. We claim that the input to D (by B) is identical to $\mathsf{emim}^A_{H_5}(v)$ in the former case and it is identical to $\mathsf{emim}^A_{H_6}(v)$ in the latter case; the proof of claim is presented shortly. Since D distinguishes the distributions with probability $1/2p(n)$, B wins the hiding game with advantage at least $1/3p(n)$.

Next, we argue that $B \in \mathcal{C}_{d_1}$: Apart from running A, B runs $o\mathcal{E}_3$ on at most $m = \mathsf{poly}(n)$ commitments $\widetilde{c3}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_4, d_4}$ and $o\mathcal{E}_3 \in \mathcal{C}^{\wedge}_{d_2, S_4}$,

$$\begin{aligned}
\mathsf{dep}(B) &= \mathsf{dep}(A) + m \cdot \mathsf{dep}(o\mathcal{E}_3) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_4) + \mathsf{poly}(n) \cdot \mathsf{poly}(d_2) \\
&< \mathsf{poly}(d_1) \qquad (\text{since, } d_1 >> d_4, d_2 \text{ from Equation (8)})
\end{aligned}$$

Furthermore, $\mathsf{size}(B) < \mathsf{poly}(S^*)$. Therefore, B belongs to the circuit class $\mathcal{C}_{d_1}$ which contradicts the $\mathcal{C}_{d_1}$-hiding of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$.

It remains to show that the input to D is identical to $\mathsf{emim}^A_{H_5}(v)$ (resp., $\mathsf{emim}^A_{H_6}(v)$), even though B uses values extracted by running $o\mathcal{E}_3$ on $\widetilde{c3}_i$ as input to D, instead of running $o\mathcal{E}_1$ on $\widetilde{c1}_i$.

For every successful right interaction i, B runs $o\mathcal{E}_3$ on $\widetilde{c3}_i$ to obtain $(\widetilde{v}'_i, \widetilde{d1}'_i)$. We claim that the value $\widetilde{v}'_i$ is identical to the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$, except with negligible probability. Since i is successful, by Claim 14 we know that with over-whelming probability A does not commit to a fake witness in right interaction i. Then by the soundness of ZAP, A must have proved that the commitments $\widetilde{c1}_i$ and $\widetilde{c3}_i$ are valid and $\widetilde{c3}_i$ commits to a decommitment of $\widetilde{c1}_i$. Therefore, by the over-extractability of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ the value $(\widetilde{v}'_i, \widetilde{d1}'_i)$ extracted from $\widetilde{c3}_i$ is identical to $\mathsf{val}(\widetilde{c3}_i)$ with over-whelming probability, where $\mathsf{val}(\widetilde{c3}_i)$ is a decommitment of $\widetilde{c1}_i$ — $(\widetilde{v}_i, \widetilde{d1}_i)$. Next, due to the over-extractability of ECom_1, the value extracted by $o\mathcal{E}_1$ from $\widetilde{c1}_i$ is identical to $\mathsf{val}(\widetilde{c1}_i) = \widetilde{v}_i$. Therefore, the value $\widetilde{v}_i$ obtained by B is identical to the value that $o\mathcal{E}_1$ extracts from $\widetilde{c1}_i$. This is now sufficient to conclude that the input to D is identical to $\mathsf{emim}^A_{H_5}(v)$ (resp., $\mathsf{emim}^A_{H_6}(v)$) when B receives a commitment to v (resp., v_0), except with negligible probability. Hence the claim holds. □

This concludes the proof of Theorem 10 and Theorem 11.

6.4 Amplifying Length of Identities

Given a tag-based commitment scheme $\langle \widehat{C}, \widehat{R} \rangle$ for $t(n)$-bit identities which is concurrent non-malleable w.r.t. commitment, Dolev, Dwork and Naor [DDN00] construct a tag-based commitment scheme $\langle \widetilde{C}, \widetilde{R} \rangle$ for exponentially larger identities, namely identities of length $2^{t(n)-1}$-bits. In their work [DDN00], they show that their transformation results in a commitment scheme that can accomodate significantly larger length of identities but degrades concurrent non-malleability w.r.t. commitment to stand-alone non-malleability w.r.t. commitment. Furthermore, their reduction also incurs a polynomial security loss.

The commitment schemes considered in this work are non-malleable w.r.t. extraction and we claim that their transformation also works for such schemes. That is, we show that if $\langle \widehat{C}, \widehat{R} \rangle$ is concurrent non-malleable w.r.t. extraction then commitment scheme $\langle \tilde{C}, \tilde{R} \rangle$ is standalone non-malleable w.r.t. extraction. The key idea towards amplifying the length of identities is embedding a $2^{t(n)-1}$-bit identity into $2^{t(n)-1}$ number of $t(n)$-bit identities — we, thereby, refer to this idea as the "log-n" trick. The protocol from [DDN00] is based on the log-n trick and is described below.

The committer $\widetilde{C}$ and receiver $\widetilde{R}$ receive the security parameter 1^n and identity $\mathsf{id} \in \{0,1\}^{t'(n)}$ as common input where $t'(n) = 2^{t(n)-1}$. Furthermore, $\widetilde{C}$ gets a private input $v \in \{0,1\}^n$ which is the value to be committed.

- *Commit stage:*

  1. To commit to a value $v \in \{0,1\}^n$, $\widetilde{C}$ chooses t' random shares $r_0, r_1, \ldots, r_{t'-1} \in \{0,1\}^n$ such that $v = r_0 \oplus r_1 \oplus \ldots \oplus r_{t'-1}$.
  2. For each $0 \leq i \leq t'-1$, $\widetilde{C}$ and $\widetilde{R}$ run $\langle \widehat{C}, \widehat{R} \rangle$ to commit to r_i (in parallel) using identity $(i, \mathsf{id}[i])$ where $\mathsf{id}[i]$ is the ith bit of id. Let d_i be the corresponding decommitment string.

Let c_i be the transcript of $\langle \widehat{C}, \widehat{R} \rangle$ committing to r_i with identity $(i, \mathsf{id}[i])$. Then we denote by $c = \{c_i\}_{i \in [t']}$ the entire transcript of the interaction.

- *Reveal stage:*
  On receiving the decommitment $(v, \{r_i\}_i, \{d_i\})$, $\widetilde{R}$ verifies that

  1. For each $i \in [t']$, c_i is a commitment to r_i using $\langle \widehat{C}, \widehat{R} \rangle$ and identity $(i, \mathsf{id}[i])$.
  2. $v = r_0 \oplus r_1 \oplus \ldots \oplus r_{t'-1}$.

  $\widetilde{R}$ accepts the decommitment iff the above conditions hold.

Furthermore, let us assume that $\langle \widehat{C}, \widehat{R} \rangle$ is over-extractable w.r.t. extractor $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ then we construct an extractor $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ for $\langle \tilde{C}, \tilde{R} \rangle$ as follows,

- Extraction - Algorithm $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$:
  On receiving $\mathsf{id} \in \{0,1\}^{t'}$ and commitment $c = \{c_i\}_{i \in [t']}$, $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ runs $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ on each c_i obtaining output r'_i. If any of the r'_i is $\perp$ then $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ outputs a $\perp$. Otherwise, it outputs $v' = r'_0 \oplus r'_1 \oplus \ldots \oplus r'_{l-1}$ as the extracted value.

Theorem 13 (Log-n trick [DDN00])**.** *Let t be such that $t'(n) = 2^{t(n)-1}$ is a polynomial. Let $\langle \widehat{C}, \widehat{R} \rangle$ be a commitment scheme and $\mathcal{C}$ be a class of circuits that is closed under composition with $\mathcal{P}$/poly.*

1. *If $\langle \widehat{C}, \widehat{R} \rangle$ is a tag based statistically binding commitment scheme for $t(n)$-bit identities then $\langle \tilde{C}, \tilde{R} \rangle$ is a tag based statistically binding commitment scheme for identities of length $t'(n) = 2^{t(n)-1}$ bits.*
2. *If $\langle \widehat{C}, \widehat{R} \rangle$ is concurrent $\mathcal{C}$-non-malleable w.r.t. commitment then $\langle \widetilde{C}, \widetilde{R} \rangle$ is one-one $\mathcal{C}$-non-malleable w.r.t. commitment.*
3. *If $\langle \widehat{C}, \widehat{R} \rangle$ is (d, S)-over-extractable by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ then $\langle \widetilde{C}, \widetilde{R} \rangle$ is (d, S)-over-extractable by $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$. Furthermore, if $\langle \widehat{C}, \widehat{R} \rangle$ is concurrent $\mathcal{C}$-non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ then $\langle \tilde{C}, \tilde{R} \rangle$ is standalone $\mathcal{C}$-non-malleable w.r.t. extraction by $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$.*

Proof. We prove each of the above in the following:

- Statistically binding and tag lengths: The statistical binding of $\langle \widetilde{C}, \widetilde{R} \rangle$ follows from the statistical binding of $\langle \widehat{C}, \widehat{R} \rangle$. Furthermore, $\langle \widetilde{C}, \widetilde{R} \rangle$ as defined above accomodates identities of length $t' = 2^{t(n)-1}$-bits.
- Non-malleability w.r.t. commitment: This is proven in [DDN00].
- Non-malleability w.r.t. extraction: This follows syntactically from the same proof for non-malleability w.r.t. commitment presented in [DDN00].

- Over-extractability: A valid commitment $c = \{c_i\}_{i\in[t']}$ is such that every c_i is a valid commitment for $\langle \widehat{C}, \widehat{R}\rangle$. Due to the over-extractability of $\langle \widehat{C}, \widehat{R}\rangle$ w.r.t. $\widehat{o\mathcal{E}}_{\mathsf{NM}}$, for every $i \in [t']$, the extractor $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ extracts the correct value r'_i except with negligible probability $\nu(n)$. Therefore, $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ extracts the correct value from c except with probability at most $t' \cdot \nu(n)$. Since, t' is a polynomial, $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ fails with negligible probability. Moreover, $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ runs $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ on t' commitments and rest of the computation takes $\mathsf{poly}(n)$ time. Therefore, if $\widehat{o\mathcal{E}}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{d,S}$ then so does $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$. Therefore, $\langle \tilde{C}, \tilde{R}\rangle$ is (d, S)-over-extractable w.r.t. $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$.

□

7 Concurrent Non-malleable Commitment for n-bit Identities

In this section, we describe the construction of a concurrent non-malleable commitment scheme $\langle C^*, R^*\rangle$ that can accomodate n-bit identities. The idea is to start with the basic commitment scheme from Section 5 that is one-one non-malleable w.r.t. extraction for short identities say $t(n)$-bits. Then apply the non-malleability strengthening technique described in Section 6.3 followed by the log-n trick [DDN00] described in Section 6.4 repeatedly until the length of the identities reaches n-bits. The resulting commitment scheme is the commitment scheme $\langle C^*, R^*\rangle$. We detail the construction of $\langle C^*, R^*\rangle$ more formally in Section 7.1. Then provide instantiations in Section 7.2 and finally discuss the efficiency of the scheme $\langle C^*, R^*\rangle$ in Section 7.3.

7.1 Commitment Scheme $\langle C^*, R^*\rangle$

We formally describe the construction of $\langle C^*, R^*\rangle$ that is concurrent non-malleable w.r.t. commitment (and extraction) for n-bit identities. As mentioned above we initially start with a commitment scheme $\langle C^0, R^0\rangle$ for $t(n)$-bit identities and apply the non-malleability strengthening and log-n trick repeatedly, for say $r(n)$ times, until we reach identities of length n-bits.

- Initial Scheme $\langle C^0, R^0\rangle$:
  The initial scheme $\langle C^0, R^0\rangle$ is the basic commitment scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ that is one-one non-malleable w.r.t. extraction for identities of length $\mathsf{id}^0(n) = t(n)$-bits. Furthermore, let $\langle C^0, R^0\rangle$ be non-malleable against circuits of depth at most $\mathsf{poly}(S^0)$ and size at most $\mathsf{poly}(S^0)$ and extractable by an extractor of depth at most $\mathsf{poly}(S'^0)$ and size at most $\mathsf{poly}(S'^0)$. [10]

- Identity Amplification Step for $r(n)$ Times:
  Next, we repeatedly apply the following two steps $r(n)$ times. Let $\langle C^{j-1}, R^{j-1}\rangle$ be the commitment scheme at the end of the $j-1$-th iteration for $j \in \{1, \ldots, r(n)\}$. We describe below the j-th iteration below. Let $\langle C^{j-1}, R^{j-1}\rangle$ be one-one non-malleable w.r.t. commitment (and extraction) for identities of length $\mathsf{id}^{j-1}(n)$-bits. Furthermore, let $\langle C^{j-1}, R^{j-1}\rangle$ be non-malleable against circuits of depth at most $\mathsf{poly}(S^{j-1})$ and size at most $\mathsf{poly}(S^{j-1})$ and extractable by an extractor of depth at most $\mathsf{poly}(S'^{j-1})$ and size at most $\mathsf{poly}(S'^{j-1})$.

  1. Non-malleability Strengthening Technique:
     First, using an appropriate hierarchy of functions as described in Equation 8, we apply

[10] Note that the initial scheme as presented in Section 5 is non-malleable against circuits of depth at most $\mathsf{poly}(d_0)$ and size at most $\mathsf{poly}(S_0)$ where $d_0 << S_0$. However, the above still implies that it is still non-malleable against circuits of depth at most $\mathsf{poly}(d_0)$ and size at most $\mathsf{poly}(d_0)$.

the non-malleability strengthening technique to $\langle C^{j-1}, R^{j-1}\rangle$ to boost the one-one non-malleability to concurrent non-malleability. The resulting scheme $\langle \widehat{C}^j, \widehat{R}^j\rangle$, therefore, is concurrent non-malleable w.r.t. commitment (and extraction) for identities of length $\mathsf{id}^{j-1}(n)$-bits.

2. Log-n Trick:
Second, we apply the log-n trick to the concurrent non-malleable scheme $\langle \widehat{C}^j, \widehat{R}^j\rangle$ to construct a one-one non-malleable commitment scheme $\langle C^j, R^j\rangle$ for identities of length $\mathsf{id}^j(n)$ such that $\mathsf{id}^j(n) = 2^{\mathsf{id}^{j-1}(n)-1}$.

- Final Scheme $\langle C^*, R^*\rangle$:
The commitment scheme $\langle C^{r(n)}, R^{r(n)}\rangle$ constructed at the end of $r(n)$ iterations is one-one non-malleable for identities of length $\mathsf{id}^{r(n)}$. We apply the non-malleability strengthening technique one more time to $\langle C^{r(n)}, R^{r(n)}\rangle$ to boost the one-one non-malleability to concurrent non-malleability. The resulting scheme is $\langle C^*, R^*\rangle$ which is concurrent non-malleable for identities of length $\mathsf{id}^{r(n)}(n)$-bits.

Note that we begin we identities of length $\mathsf{id}^0 = t(n)$ and identities in successive iterations satisfy the following,

$$\mathsf{id}^j(n) = 2^{\mathsf{id}^{j-1}(n)-1} .$$

Then it is easy to see that for $\mathsf{id}^{r(n)}(n) \geq n$, we need to apply the identity amplification step $r(n) = O(\log^* n - \log^* t(n))$ times.

7.2 Instantiations

The initial scheme constructed in Section 5 and the identity amplification step described in Sections 6.3,6.4 require a family of depth-robust and size-robust commitment schemes, and a family of non-uniform collision resistant hash functions which are based on some hierarchy of non-decreasing functions. Below we detail the size of this hierarchy required for constructing $\langle C^*, R^*\rangle$ from the initial scheme $\langle C^0, R^0\rangle$ for $t(n)$-bit identities and $r(n)$ iterations of the identity amplification step. Then we give instantiations of this hierarchy firstly from sub-exponential security and then from the strictly weaker sub-subexponential security.

Initial Scheme $\langle C^0, R^0\rangle$: We start with the basic commitment scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ for $t(n)$-bit identities. As described in Section 5, the construction of $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ for $t(n)$-bit identities requires a family of $2^{t(n)}$ size-robust and depth-robust commitment schemes w.r.t. the following hierarchy of non-decreasing functions,

$$n << d_0 << d_1 << \ldots << d_{l-1} << d_l << S_0 << S_1 << \ldots << S_{l-1} << S_l ,$$

where $l = 2^{t(n)}$ such that for every $i \in \{0,1\}^{t(n)}$,

- there exists a depth-robust commitment scheme $(\mathsf{ECom}_{d_i}, \mathsf{EOpen}_{d_i})$ that is $\mathcal{C}_{d_i}$-hiding and (d_{i+1}, d_{i+1})-over-extractable w.r.t. an extractor $o\mathcal{E}_{d_i}$.

- there exists a size-robust commitment scheme $(\mathsf{ECom}_{S_i}, \mathsf{EOpen}_{S_i})$ that is $\mathcal{C}_{S_i}$-hiding and $(\mathsf{poly}(n), S_{i+1})$-over-extractable w.r.t. an extractor $o\mathcal{E}_{S_i}$.

Therefore, to construct the initial commitment scheme we need a hierarchy of $2(l+1) = 2(2^{t(n)}+1)$ non-decreasing functions.

Identity Amplification Step: Consider the $j+1$-th iteration of the identity amplification step described in the construction of $\langle C^*, R^* \rangle$. In the $j+1$-th iteration, we are applying the strengthening technique to the commitment scheme $\langle C^j, R^j \rangle$ which is $\mathcal{C}^{\wedge}_{S^j,S^j}$-non-malleable and extractable by a circuit of size at most $\mathsf{poly}(S'^j)$. The strengthening technique requries a family of four depth-robust [11] and four size-robust commitment schemes. Furthermore, it also requires a family of non-uniform collision-resistant hash functions w.r.t. the following hierarchy of non-decreasing functions,

$$n << d_4^j << d_3^j << d_2^j << d_1^j << S_2^j << S_{\mathsf{CRH}}^j << S_{\mathsf{CRH}}'^j << S^j << S'^j << S_3^j << S_4^j << S_4'^j << S^* ,$$

such that,

- $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ is a perfectly binding depth-robust commitment scheme which is $\mathcal{C}_{d_1^j}$-hiding and (S_2^j, S_2^j)-over-extractable w.r.t. extractor $o\mathcal{E}_1$.
- $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_2^j,S_2^j}$-hiding and $(d_1^j, S_{\mathsf{CRH}}^j)$-over-extractable w.r.t. extractor $o\mathcal{E}_2$.
- $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_3^j,S_3^j}$-hiding and $(d_2^j, S_4'^j)$-over-extractable w.r.t. extractor $o\mathcal{E}_3$.
- $(\mathsf{ECom}_4, \mathsf{ECom}_4)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_4^j,S_4^j}$-hiding and $(d_3^j, S_4'^j)$-over-extractable w.r.t. extractor $o\mathcal{E}_4$.
- $\mathcal{H} = \{D_n\}$ is a $\mathcal{C}_{S_{\mathsf{CRH}}}$-collision-resistant family of hash functions such that a collision can be found by a circuit in $\mathcal{C}_{S'_{\mathsf{CRH}}}$.

Furthermore, we apply the log-n trick to the resulting commitment scheme. Note that the log-n trick does not rely on any additional tools. Therefore, in an iteration of the identity amplification step, we need four depth-robust[11], four size-robust commitment schemes and a hash function family. In other words, we need an additional at most ten non-decreasing functions per iteration. Therefore, over $r(n)$ iterations, we will need a hierarchy of $10r(n)+10$ functions. [12]

Therefore, to construct the commitment scheme $\langle C^*, R^* \rangle$ from $\langle C^0, R^0 \rangle$ for $t(n)$-bit identities, we need a hierarchy of $L = 2^{t(n)+1} + 10r(n) + 12$ non-decreasing functions, where $r(n) = O(\log^* n - \log^* t(n))$. Furthermore, L is minimized when $t(n) = O(1)$, implying $r(n) = O(\log^* n)$ and $L = O(\log^* n)$. Next, we show two approaches to instantiate a hierarchy of $L = O(\log^* n)$ non-decreasing functions, one from sub-exponential security and another from sub-subexponential security.

Instantiation from Sub-exponential Security: As mentioned above, we need to instantiate a hierarchy of L non-decreasing functions for constructing $\langle C^*, R^* \rangle$. Let the required hierarchy be the following,

$$p_1 << p_2 << \ldots << p_L . \tag{13}$$

Let $\mathcal{H}(\lambda)$ be some non-decreasing function defined on $\mathbb{N}$. First we will instantiate the hierarchy based on the existence of $2^{\mathcal{H}(\lambda)}$-secure OWPs, TL puzzles and collision-resistant hash functions.

[11] Note that the transformation actually requires three depth-and-size robust commitment schemes and a depth-robust commitment scheme but as described in Section 4.3 depth-and-size robust commitment scheme can be constructed from a single depth-robust and a size-robust commitment scheme.

[12] The additional ten functions is due an extra application of the non-malleability strengthening to boost the non-malleability of $\langle C^{r(n)}, R^{r(n)} \rangle$.

Then provide concrete parameters for the special case of sub-exponential security, that is, for $\mathcal{H} = \lambda^{\varepsilon}$ for some $\varepsilon < 1$.

We instantiate the above hierarchy from $2^{\mathcal{H}}(\lambda)$-security by varying the security parameter λ. Consider the following sequence of security parameters relationships between whom is discussed shortly,

$$n_0 \ , n_1 \ , \ldots \ , n_L \ .$$

We set the i-th level in the hierarchy p_i where $i \in \{1, \ldots, L\}$,

$$p_i = 2^{\mathcal{H}(n_i)} \ .$$

We expect the functions in the hierarchy to satisfy certain constraints in order for us to be able to instantiate the required depth-robust and size-robust commitment schemes from them. We enlist the properties below.

1. Since we expect all our primitives to be secure against any poly-sized circuit, we require that the first security parameter n_0 be such that $2^{\mathcal{H}(n_0)} \geq 2^{\omega(\log n)}$ that is,

$$\begin{aligned} \mathcal{H}(n_0) &= \omega(\log n) \ , \\ n_0 &= \mathcal{H}^{-1}(\omega(\log n)) \ . \end{aligned}$$

2. For any i, we need to be able to instantiate the following primitives,

   (a) (p_i, p_{i+1})-depth-robust commitment scheme: We instantiate such a scheme from TL puzzles by sampling puzzles with security parameter n_i.

   (b) (p_i, p_{i+1})-size-robust commitment scheme: We instantiate such a scheme from OWPs by instantiating the OWP with security parameter n_i.

   (c) (p_i, p_{i+1})-collision-resistant hash function family: A (p_i, p_{i+1})-collision-resistant hash function family is a family of hash functions that is $\mathcal{C}_{p_i}$-collision resistant and for which there exists a circuit in $\mathcal{C}_{p_{i+1}}$ that finds collisions with probability 1. We instantiate such a family by setting the security parameter for $\mathcal{H}$ as n_i, where $\mathcal{H}$ is a family of non-uniform $2^{\mathcal{H}(n)}$-collision-resistant hash functions.

   Therefore, in each of the above three cases, we require that the following relation hold between p_i and p_{i+1}.

$$\mathsf{poly}(p_i) = \mathsf{poly}(2^{\mathcal{H}(n_i)}) < 2^{n_i} \leq p_{i+1} = 2^{\mathcal{H}(n_{i+1})} \ .$$

   The above constraint implies the following relation between the security parameters of adjacent levels.

$$n_{i+1} = \mathcal{H}^{-1}(n_i) \ = \left(\mathcal{H}^{-1}\right)^{i+1}(n_0) = \left(\mathcal{H}^{-1}\right)^{i+2}(\omega(\log n)) \ .$$

   Therefore the ith security parameter n_i is,

$$n_i = \left(\mathcal{H}^{-1}\right)^{i+1}(\omega(\log n)) \ .$$

3. Finally we require that the last security parameter n_L be upper-bounded by some $\mathsf{poly}(n)$,

$$n_L = \left(\mathcal{H}^{-1}\right)^{L+1}(\omega(\log n)) \leq \mathsf{poly}(n) \ . \tag{14}$$

Now let us consider the case of sub-exponential security, that is, let $\mathcal{H} = \lambda^{\varepsilon}$ for some $\varepsilon < 1$. Since $\mathcal{H}$ is non-decreasing it is invertible and $\mathcal{H}^{-1}(y) = y^{1/\varepsilon}$. For the last security level n_L to be polynomially bounded, we require that,

$$(\omega(\log n))^{(1/\varepsilon)^{L+1}} \leq \mathsf{poly}(n) \ .$$

It is easy to see that from subexponential security, we can derive $L = \Theta(\log\log n)$ levels. Recall that to construct $\langle C^*, R^* \rangle$ we need $O(\log^* n)$ levels in the hierarchy, hence the above hierarchy finds an instantiation from subexponential security.

However, for our transformation, we require only $L = O(\log^* n)$ levels which is significantly less than $\Theta(\log\log n)$ levels that can be extracted from sub-exponential security. Hence, there is hope to instantiate the hierarchy from weaker than sub-exponential security. Infact, such a hierarchy can, indeed, be instantiated from strictly weaker security — *sub-subexponential* security — which we show below.

Instantiation from Sub-subexponential Security: First we define the notion of sub-subexponential security and then provide an instantiation of the hierarchy. Informally, a $2^{\mathcal{H}(\lambda)}$-secure primitive is sub-subexponential -secure if

$$\mathcal{H}(\lambda) \in \lambda^{o(1)} \ .$$

A candidate for $\mathcal{H}$ for sub-subexponential security is the following,

$$\mathcal{H}(\lambda) = \lambda^{\frac{1}{\mathcal{X}(\lambda)}} \ ,$$

where $\mathcal{X}(\lambda) = \omega(1)$ be some non-decreasing function on $\mathbb{N}$.

We ask how large (if at all) such an $\mathcal{X}(\lambda) = \omega(1)$ can be so that we can still instantiate the above hierarchy. The only point of concern is bounding the security parameter n_L of the last level, that is, we ask how large $\mathcal{X}(\lambda)$ be such that for $\mathcal{H}(\lambda) = \lambda^{\frac{1}{\mathcal{X}(\lambda)}}$ and $L = O(\log^* n)$ the following holds,

$$n_L = \left(\mathcal{H}^{-1}\right)^L (\omega(\log n)) \leq \mathsf{poly}(n) \ .$$

However the above closed form is hard to analyse so we restrict the right hand side to be n instead of a generic $\mathsf{poly}(n)$, that is,

$$\left(\mathcal{H}^{-1}\right)^L (\omega(\log n)) \leq n \tag{15}$$

Applying $\mathcal{H}$ on both sides we get,

$$\left(\mathcal{H}^{-1}\right)^{L-1} (\omega(\log n)) \leq \mathcal{H}(n) \ , \tag{16}$$

Let $n' = \mathcal{H}(n) = n^{\frac{1}{\mathcal{X}(n)}} < n$. We have,

$$\mathcal{H}(n') = \left(n'\right)^{\frac{1}{\mathcal{X}(n')}} = (\mathcal{H}(n))^{\frac{1}{\mathcal{X}(n')}} \ .$$

Since $\mathcal{X}$ is a non-decreasing function we have,

$$\mathcal{H}(n') = (\mathcal{H}(n))^{\frac{1}{\mathcal{X}(n')}} > (\mathcal{H}(n))^{\frac{1}{\mathcal{X}(n)}} \ , \tag{17}$$

Applying again $\mathcal{H}$ on both sides of Equation (16),

$$\left(\mathcal{H}^{-1}\right)^{L-2}(\omega(\log n)) \leq \mathcal{H}(n') \ , \tag{18}$$

Therefore by Equation (17) we know that as long as the following holds, Equation (18) holds.

$$\left(\mathcal{H}^{-1}\right)^{L-2}(\omega(\log n)) \leq \mathcal{H}(n)^{\frac{1}{\mathcal{X}(n)}} = n^{\frac{1}{\mathcal{X}(n)}^2} \ .$$

After repeatedly applying $\mathcal{H}$, it is easy to see that as long as the following holds, Equation 15 holds.

$$\omega(\log n) \leq n^{\frac{1}{\mathcal{X}(n)^L}} \ .$$

Furthermore, the if the following holds then the above Equation holds,

$$\mathcal{X}(n)^L \leq \frac{\log n}{\omega(\log\log n)}$$

$$\mathcal{X}(n) \leq \left(\frac{\log n}{\omega(\log\log n)}\right)^{\frac{1}{O(\log^* n)}}$$

Finally, as long as the following holds for some $c > 0$ then Equation (15) holds.

$$\mathcal{X}(n) \leq (\log^c n)^{\frac{1}{O(\log^* n)}}$$

$$\mathcal{X}(n) \leq (\log n)^{\frac{1}{\Theta(\log^* n)}} \tag{19}$$

For $\mathcal{X}(n) = \log\log n$, it is easy to see that Equation 19 holds and hence Equation (15) holds. Therefore we can instantiate the above hierarchy from $2^{n^{\frac{1}{\log\log n}}}$-secure OWPs, TL puzzles and CRHs which is strictly weaker than assuming 2^{n^ε}-security.

7.3 Efficiency of $\langle C^*, R^*\rangle$

As described in Section 7.1, to construct the scheme $\langle C^*, R^*\rangle$ we apply the identity amplification step — non-malleability strengthening technique followed by the log-n trick — $O(\log^* n)$ times. Suppose that the identity amplification step incurrs a polynomial overhead, that is, on input a scheme with computational complexity $\tau(n)$, it outputs a scheme with computational complexity $p(\tau(n))$ for some fixed polynomail p. Applying this step for a super-constant number of times leads to a scheme $\langle C^*, R^*\rangle$ with super-polynomial computational complexity.

Unfortunately, our non-malleability strengthening technique presented in Section 6 indeed incurrs polynomial overhead. Recall that on input a non-malleable commitment $\langle C, R\rangle$, the technique produces an output scheme $\langle \widehat{C}, \widehat{R}\rangle$ which uses ZAP to prove a statement that involves verifying the decommitment to a commitment of $\langle C, R\rangle$. Therefore, if the decommiment function $\mathsf{Open}(c, v, d)$ of $\langle C, R\rangle$ has complexity $\tau_{\mathsf{Open}}(n)$, the output scheme has complexity at least $p_{\mathsf{ZAP}}(\tau_{\mathsf{Open}}(n))$, where p_{ZAP} is the polynomial overhead induced by ZAP.

We show below that a simple modification can fix the problem. (We chose to present the strengthening technique in simpler terms earlier for ease of exposition.) Towards this, we introduce a new property called *open-decomposability* for commtiment schemes. We say that a scheme $\langle C, R\rangle$ is g-open-decomposable, if it is the case that, its decommitment function $\mathsf{Open}(c, v, d)$ that can be decomposed into two functions of the following form:

- a "public" function $\mathsf{PubOpen}(c)$ that can be verified without the decommitment (v, d), and
- a "private" function $\mathsf{PrivOpen}(c^*, v, d)$ that depends on the decommitment and only a small part $c^* = \pi(c)$ of the commitment c, and takes polynomial time $g(n)$.

Open accepts if and only if both $\mathsf{PubOpen}$ and $\mathsf{PrivOpen}$ accepts. Consider applying the non-malleability strengthening technique on such a g-open-decomposable commitment scheme. Instead of using ZAP to verify whether Open accepts, it is equivalent to verify whether $\mathsf{PubOpen}$ accepts in the clear (outside ZAP) and only verifies whether $\mathsf{PrivOpen}$ accepts using ZAP. This simple change reduces the overhead induced by the ZAP proof from $p_{\mathsf{ZAP}}(\tau_{\mathsf{Open}}(n))$ to $p_{\mathsf{ZAP}}(g(n))$. Our key observation is that the initial non-malleable schemes, as well as all intermediate schemes produced throughout the iterations, are all open-decomposable w.r.t. small polynomials. Based on this observation and careful analysis, we can show that the complexity of the final scheme is polynomially bounded.

Open-decomposability We now formally define the notion of open-decomposability.

Definition 19 (g-open-decomposability)**.** *Let g be a polynomial. We say that a commitment scheme $\langle C, R\rangle$ is g-open-decomposable if there exist efficiently computable functions* $\mathsf{PubOpen}$, $\mathsf{PrivOpen}$, *and π, such that,*

for all $n \in Z$, and $c \in \{0,1\}^{m(n)}$, $v \in \{0,1\}^n$ and $d \in \{0,1\}^{l(n)}$,

$$\mathsf{Open}(c, v, d) = 1 \iff \mathsf{PubOpen}(c) = 1 \wedge \mathsf{PrivOpen}(c^* = \pi(c), v, d) = 1 \ ,$$

where $\mathsf{PrivOpen}$ *runs in time $g(n)$.*

Above, $m(n)$ and $l(n)$ are respectively the maximal lengths of commitments and decommitments generated using $\langle C, R\rangle$ with security parameter n.

Using the above notion, we next describe the modified non-malleability strengthening technique and log-n trick. We analyze the open-decomposability property of the schemes produced by iteratively applying these two transformations to the initial schemes constructed in Section 5, and show that the growth of the complexity of these schemes is polynomially bounded.

More specifically, let g be a sufficiently large polynomial that, in particular, is larger than the complexity of all depth-and-size robust commtiment schemes, ECom's, used for constructing the initial schemes and in the transformations. By the analysis in Section 7.2, all the ECom's used have polynomial complexity. This implies that the initial non-malleable commitment schemes (consisting of invokation of two ECom schemes) does satisfy g-open-decomposability (by simply setting $\mathsf{PubOpen}$ to the constant function outputting 1 and $\mathsf{PrivOpen} = \mathsf{Open}$ itself). Then, we show that the non-malleability stengthening technique always outputs a scheme that is g-open-decomposable, and on input such a scheme, the log-n trick produces a scheme that is $h(n)$-open-decomposable for $h(n) = ng(n)$.

Modification to the strengthening technique described in Section 6.3: Let $\langle C, R\rangle$ be one-one non-malleable w.r.t. extraction and satisfy h-open-decomposable w.r.t. $(\mathsf{PubOpen}, \mathsf{PrivOpen}, \pi)$. We describe the changes (highlighted in red) to the non-malleability strengthening technique.

- Commit stage - First round: Same as before.
- Commit stage - Second round: Steps 1, 2 and 4 are same as before.

3. Given a_{ZAP}, $\widehat{C}$ computes the second message b_{ZAP} of ZAP to prove the following OR-statement:
   (a) *either* there exists a string $\bar{v}$ such that $c1$ is a commitment to $\bar{v}$ and $c3$ commits to a decommitment of $c1$.
   (b) *or* there exists a string $\bar{s} = (x_1, x_2)$, such that,
      – $h(x_1) = h(x_2)$,
      – $c2$ is a commitment to $\bar{s}$,
      – $c4$ commits to a decommitment of $c2$,
      – $\mathsf{PrivOpen}$ accepts $(c^*, d4, v4)$ for $c^* = \pi(a_{\mathsf{NM}}, b_{\mathsf{NM}})$, and $(d4, v4)$ is a valid decommtiment to $c4$.

   $\widehat{C}$ proves the statement (a) by using a decommitment of $c3$ to $(v, d1)$ — decommitment of $c1$ to v — as the witness.

Denote by $(\hat{a}_{\mathsf{NM}}, \hat{b}_{\mathsf{NM}})$ the produced commitment.

- Reveal stage - Function $\widehat{\mathsf{Open}}((\hat{a}_{\mathsf{NM}}, \hat{b}_{\mathsf{NM}})), d1, v)$:
  Parse $(\hat{a}_{\mathsf{NM}}, \hat{b}_{\mathsf{NM}})$ and let $(a_{\mathsf{ZAP}}, b_{\mathsf{ZAP}})$, $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$, and $c1$ be the ZAP proof, the commitment of $\langle C, R\rangle$, and the ECom_1 commtiment contained in it. Accept if and only if the following functions both accept.
  – $\widehat{\mathsf{PubOpen}}(\hat{a}_{\mathsf{NM}}, \hat{b}_{\mathsf{NM}})$ accepts iff the ZAP proof $(a_{\mathsf{ZAP}}, b_{\mathsf{ZAP}})$ is accpeting and that $\mathsf{PubOpen}((a_{\mathsf{NM}}, b_{\mathsf{NM}})) = 1$.
  – $\widehat{\pi}(\hat{a}_{\mathsf{NM}}, \hat{b}_{\mathsf{NM}}) = c1$ and $\widehat{\mathsf{PrivOpen}}(c1, v, d1)$ accepts iff $\mathsf{EOpen}_1(c1, v, d1) = 1$.

The commitment scheme $\langle \widehat{C}, \widehat{R}\rangle$ is open-decomposable w.r.t. $(\widehat{\mathsf{PubOpen}}, \widehat{\mathsf{PrivOpen}}, \widehat{\pi})$. Since $\widehat{\mathsf{PrivOpen}}$ only checks the decommitment of the ECom_1 commitment, its runtime is bounded by $g(n)$. Therefore, $\langle \widehat{C}, \widehat{R}\rangle$ satisfies $g(n)$-open-decomposability. On the other hand, since $\mathsf{PrivOpen}$ has complexity $h(n)$, the ZAP proof incurrs an additive $\mathsf{poly}(n, g(n), h(n))$ overhead. Thus, if the computational complexity of $\langle C, R\rangle$ is $cc(n)$, the computational complexity $\widehat{cc}(n)$ of $\langle \widehat{C}, \widehat{R}\rangle$ is:

$$\widehat{cc}(n) = cc(n) + \mathsf{poly}(n, g(n), h(n))$$

Modification to log-n trick described in Section 6.4: Let $\langle \widehat{C}, \widehat{R}\rangle$ be concurrent non-malleable (w.r.t. commitment and extraction) for $l(n)$-bit identities, and be $g(n)$-open-decomposable w.r.t. $(\widehat{\mathsf{PubOpen}}, \widehat{\mathsf{PrivOpen}}, \widehat{\pi})$. The log-n trick results in a commitment scheme $\langle \widetilde{C}, \widetilde{R}\rangle$ which is one-one non-malleable (w.r.t. commitment and extraction) for identities of length $l'(n) = 2^{l(n)-1} < n$. We show that $\langle \widetilde{C}, \widetilde{R}\rangle$ is $h(n)$-open-decomposable w.r.t. $(\widetilde{\mathsf{PubOpen}}, \widetilde{\mathsf{PrivOpen}}, \widetilde{\pi})$ described below.

- Commit stage: Same as before.
  Let $\widetilde{a}_{\mathsf{NM}}, \widetilde{b}_{\mathsf{NM}}$ be the produced commitment, which contains l' commitments of $\langle \widehat{C}, \widehat{R}\rangle$, denoted as $\left\{\widehat{a}^i_{\mathsf{NM}}, \widehat{b}^i_{\mathsf{NM}}\right\}_{i\in[l']}$.

- Reveal stage - Function $\widetilde{\mathsf{Open}}((\tilde{a}_{\mathsf{NM}}, \tilde{b}_{\mathsf{NM}})), d, v)$: Accept if and only if the following functions both accept.
  – $\widetilde{\mathsf{PubOpen}}$ accepts iff for every i, $\widehat{\mathsf{PubOpen}}(\widehat{a}^i_{\mathsf{NM}}, \widehat{b}^i_{\mathsf{NM}})$ accepts.

– $\widetilde{\pi}(\widetilde{a}_{\mathsf{NM}}, \widetilde{b}_{\mathsf{NM}}) = \left\{ c_i^* = \widehat{\pi}(\widehat{a}^i_{\mathsf{NM}}, \widehat{b}^i_{\mathsf{NM}}) \right\}_i$ and $\widetilde{\mathsf{PrivOpen}}$ accepts iff for every i, $\widehat{\mathsf{PrivOpen}}$ accpets c_i^* w.r.t. d, v.

Note that the running time of $\widetilde{\mathsf{PrivOpen}}$ is at most $l'(n) \cdot g(n) \leq h(n)$, and hence $\langle \widetilde{C}, \widetilde{R} \rangle$ is $h(n)$-open-decomposable. Furthermore, if the computational complexity of $\langle \widehat{C}, \widehat{R} \rangle$ is $\widehat{cc}(n)$, the computational complexity of $\langle \widetilde{C}, \widetilde{R} \rangle$ is bounded by $l'(n)\widehat{cc}(n)$.

Putting Pieces Together Every iteration, say the j'th, starts with a scheme $\langle C^j, R^j \rangle$ that is $h(n)$-open-decomposable (the inital schemes are g-open-decomposable). Applying the non-malleability strengthening technique produces a scheme $\langle \widehat{C}^j, \widehat{R}^j \rangle$ that is is $g(n)$-open-decomposable. Following that, the log-n trick produces a scheme $\langle C^{j+1}, R^{j+1} \rangle$ that is $h(n)$-open-decomposable for $h(n) = ng(n)$. Furthermore, Let $cc(j)$ denote the computational complexity of scheme $\langle C^j, R^j \rangle$. Then we have:

$$\begin{aligned}
cc(j+1) &= \mathsf{id}^{j+1}(n)\left(cc(j) + \mathsf{poly}(n, g(n), h(n))\right) \\
&= \mathsf{id}^{j+1}(n)\left(\mathsf{id}^{j}(n)(cc(j-1) + \mathsf{poly}(n)) + \mathsf{poly}(n)\right) \\
&\leq \mathsf{id}^{j+1}(n)\mathsf{id}^{j}(n)cc(j-1) + \mathsf{id}^{j+1}(n)\mathsf{id}^{j}(n)\mathsf{poly}(n) + \mathsf{id}^{j+1}(n)\mathsf{poly}(n) \\
&\leq \mathsf{id}^{j+1}(n)\mathsf{id}^{j}(n)cc(j-1) + 2\mathsf{id}^{j+1}(n)\mathsf{id}^{j}(n)\mathsf{poly}(n) \\
&\leq \prod_{1 \leq k \leq j+1} \mathsf{id}^k(n)cc(0) + (j+1)\left(\prod_{1 \leq k \leq j+1} \mathsf{id}^k(n)\right)\mathsf{poly}(n)
\end{aligned}$$

Since the total number of iterations is $O(\log^* n)$ and the lengths of identities grow exponentially fast, we have that the running time of the final scheme $\langle C^*, R^* \rangle$ is upper-bounded by a polynomial.

8 Non-interactive Non-Malleable Commitment against Uniform Adversaries

In this section, we show that when restricting attention to uniform attackers, the first message in our 2-round concurrent non-malleable commitment scheme constructed in Section 7 can be removed. Recall that these 2-round protocols are obtained by iteratively applying the amplification transformation in Section 6 to the basic schemes for short identities in Section 5. While the basic schemes are in fact non-interactive, the amplification technique, however, produces schemes with 2 rounds. Our amplification technique involves two steps: Applying the DDN $\log n$ trick, which is actually round preserving, and the security strengthening step that lifts one-one non-malleability w.r.t. extraction to concurrent non-malleability w.r.t. extraction and commitment, while preserving the length of identities. In the security strengthening step, the output scheme has two rounds, where the first message is sent by the receiver and contains the index of a randomly sampled function h from a family of non-uniform CRHFs, the first message of a ZAP proof, and the first message of the input non-malleable commitment scheme (if there is any). Therefore, to remove the first message, our idea is to simply replace h for a fixed uniform CRHF, and replace ZAP with a NIWI, so that the transformation when applied to a non-interactive input commitment scheme, produces a non-interactive output scheme. The only drawback is that with the use of uniform CRHF, the output scheme is only secure against uniform adversaries.

Below, we first adapt the notions of non-malleability w.r.t. extraction and commitment to the setting of uniform attackers, and then describe the new amplification step.

8.1 Non-malleability against Uniform Adversaries

Non-malleability w.r.t. commitment (or w.r.t. extraction) against *uniform* attackers are defined identically to that against non-uniform attackers as in Definition 15 (or Definition 16 resp.) in Section 3.5, except from one difference in the man-in-the-middle execution. Recall that in the non-uniform case, the man-in-the-middle attacker A receives in the left interactions commitments to arbitrary values $v_1, \cdots, v_m$, and non-malleability requires the view of A and the values committed in (or extracted from) the right interactions to be indistinguishable, no matter A receives commitments to one set of *arbitrary* values $v_1^0, \cdots, v_m^0$ or another $v_1^1, \cdots, v_m^1$.

In the uniform case, letting a uniform attacker A' receive commitments to arbitrary values is similar to giving it non-uniform advices. Therefore, we modify the man-in-the-middle execution to let the attacker choose the challenge messages (v_i^0, v_i^1) it wishes to receive commitments to in the i'th left commtiment before that interaction starts. Moreover, every man-in-the-middle execution is parameterized with a bit $b \in \{0,1\}$, and commits to $\{v_i^b\}$ in left interactions — let $\mathsf{uMIM}^A_{\langle C,R\rangle}(1^n, b)$ denote such an execution. We further denote by $\mathsf{umim}^A_{\langle C,R\rangle}(1^n, b)$ (or $\mathsf{uemim}^A_{\langle C,R\rangle}(1^n, b)$ resp.) the random variable describing the view of A together with the values committed in (or extracted from resp.) the right interactions.

Definition 20 (Non-malleability)**.** *A tag-based commitment scheme $\langle C, R\rangle$ is said to be concurrent T-non-malleable against uniform attackers if for every $\mathsf{poly}(T)$-time uniform Turing machine A participating in $m = \mathsf{poly}(n)$ concurrent interactions, the following ensembles are computationally indistinguishable:*

$$\left\{\mathsf{umim}^A_{\langle C,R\rangle}(1^n, 0)\right\}_{n\in\mathbb{N}} ,$$
$$\left\{\mathsf{umim}^A_{\langle C,R\rangle}(1^n, 1)\right\}_{n\in\mathbb{N}} .$$

Moreover, it is said to be concurrent T-non-malleable w.r.t. extraction against uniform attackers, if the above indistinguishability holds for $\mathsf{uemim}^A_{\langle C,R\rangle}(1^n, 0)$ and $\mathsf{uemim}^A_{\langle C,R\rangle}(1^n, 1)$.

8.2 1-Message Security Strengthening Technique

We now present our one-message transformation for security strengthening. For some hierarchy of non-decreasing functions on $\mathbb{N}$ satisfying,

$$\begin{aligned} &n << d_4 << d_3 << d_2 << d_1 << S_2 << S_{\mathsf{CRH}} << \\ &S'_{\mathsf{CRH}} << S_{\mathsf{NM}} << S'_{\mathsf{NM}} << S_3 << S_4 << S'_4 << S^* , \end{aligned} \tag{20}$$

the transformation relies on the following building blocks:

1. $(\mathsf{oNICom}, \mathsf{oNIOpen})$ is a non-interactive, tag-based commitment scheme for $t(n)$-bit identities that is S'_{NM}-over-extractable by extractor $o\mathcal{E}_{\mathsf{NI}}$. Furthermore, $\langle C, R\rangle$ is one-one S_{NM}-non-malleable w.r.t. extraction by $o\mathcal{E}_{\mathsf{NI}}$ against uniform adversaries.
2. $\{(\mathsf{ECom}_i, \mathsf{EOpen}_i)\}_{1\le i\le 4}$ are identical to that in Section 6.3.
3. NIWI is a non-interactive $\mathcal{C}_{S^*}$-witness-indistinguishable proof.
4. $H = \{H_n\}_n$ is a S_{CRH}-uniform-collision resistant hash function such that there exists a $\mathsf{poly}(S'_{\mathsf{CRH}})$-time TM which finds collisions for $\mathcal{H}$ with probability 1.

Using the above mentioned buiding blocks, the transformation produces $(\mathsf{cNICom}, \mathsf{cNIOpen})$ which is non-interactive, tag-based commitment scheme for $t(n)$-bit identities that is S_2-over-extractable w.r.t. an extractor $\widehat{o\mathcal{E}}_{\mathsf{NI}}$. Furthermore, $(\mathsf{cNICom}, \mathsf{cNIOpen})$ is both concurrent d_4-non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}}_{\mathsf{NI}}$ and concurrent d_4-non-malleable (w.r.t. commitment) against uniform attackers.

The committer $\widehat{C}$ and the receiver $\widehat{R}$ receive the security parameter 1^n and identity $\mathsf{id} \in \{0,1\}^{t(n)}$ as common input. Furthermore, $\widehat{C}$ gets a private input $v \in \{0,1\}^n$ which is the value to be committed.

- Commit stage:

  1. $\widehat{C}$ computes a commitment $c1$ to the value v using ECom_1. Let $d1$ be the corresponding decommitment string.
  2. $\widehat{C}$ computes a commitment $c3$ to the decommitment $(v, d1)$ of $c1$ using ECom_3.
  3. $\widehat{C}$ computes a commitment $c2$ to a random string $r1$ using ECom_2.
  4. $\widehat{C}$ computes a commitment $c\mathsf{NM}$ to a random string $r3$ using oNICom using identity id.
  5. $\widehat{C}$ computes a commitment $c4$ to a random string $r3$ using ECom_4.
  6. $\widehat{C}$ computes the NIWI proof π to prove the following OR-statement:
     (a) *either* there exists a string $\bar{v}$ such that $c1$ is a commitment to $\bar{v}$ and $c3$ commits to a decommitment of $c1$.
     (b) *or* there exists a string $\bar{s} = (x_1, x_2)$ such that $c2$ is a commitment to $\bar{s}$, $c4$ commits to a decommitment of $c2$, $c\mathsf{NM}$ commits to a decommitment of $c4$ and $\mathsf{CRH}_n(x_1) = \mathsf{CRH}_n(x_2)$.

     $\widehat{C}$ proves the statement (a) by using a decommitment of $c3$ to $(v, d1)$ — decommitment of $c1$ to v — as the witness.
  7. $\widehat{C}$ sends $(c1, c2, c3, c4, c\mathsf{NM}, \pi)$ as commitment to $\widehat{R}$ and keeps the decommitment $(v, d1)$ private.

- Reveal stage:
  On receiving $(v, d1)$ from $\widehat{C}$, $\widehat{R}$ accepts the decommitment if the NIWI proof is accepting and if $\mathsf{EOpen}_1(c1, v, d1) = 1$. Otherwise, it rejects.

- Extraction - Extractor $\widehat{o\mathcal{E}}_{\mathsf{NI}}$:
  On receiving a commitment c and identity id, $\widehat{o\mathcal{E}}_{\mathsf{NI}}$ first verifies the NIWI proof and outputs $\perp$ if the proof is not accepting. Otherwise, it runs the extractor $o\mathcal{E}_1$ on $c1$ and outputs the extracted value v'.

Theorem 14. *$\langle \widehat{C}, \widehat{R} \rangle$ is a non-interactive, perfectly binding, (S_2, S_2)-over-extractable commitment scheme for identities of length $t(n)$. Furthermore, it is concurrent $\mathcal{C}^{\wedge}_{d_4,d_4}$-non-malleable (w.r.t. commitment) and non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ against uniform adversaries.*

It is easy to see that $\langle \widehat{C}, \widehat{R} \rangle$ is perfectly binding and (S_2, S_2)-over-extractable. The non-malleability properties follow syntactically from the same proof as that of Theorem 10 and 11 w.r.t. the 2-round security strengthening technique in Section 6.3. The only slight difference is that when reducing to the collision resistance of the hash function, and the non-malleability w.r.t. extraction of the input commitment scheme, we need to ensure that the reduction is a uniform Turing machine, which can be done easily. More specifically, in Section 6.3,

- we rely on the collision resistance of hash functions in order to show that Invariant 2 holds in hybrid $H_0(v)$ (Claim 3), and

- we rely on the non-malleability w.r.t. extraction of the input commitment scheme in order to show that Invariant 2 holds in $H_3(v)$ (Claim 8) and that the emim random variable is indistinguishable in $H_2(v)$ and $H_3(v)$ (Claim 9).

We now observe that the reductions presented in the proof of Claim 3, 8 and 9 can be made uniform. First, these reductions run internally 1) the adversary, 2) the extractors for different commitment schemes, 3) possibly a strategy for finding collisions (for the second bullet point), and some other computations, all of which can be implemented using uniform Turing machines. Furthermore, these reductions have two values hardwired in — the value v committed to in the left and the index k of a "special" right interaction. When adapting to the uniform setting, since the attacker chooses the two challenge messages v_0, v_1 to be committed to in the left interaction, the reduction only need to have a single bit b hardwired in. Furthermore, since there are only a polynomial number of right interactions, instead of hard-wiring k, the reduction can simply guess k at random, at the cost of losing a factor of k in its advantage. Therefore, by essentially the same proof, we can show the same in the uniform setting. We hence omit the complete proof.

References

[Bar02] Boaz Barak. Constant-round coin-tossing with a man in the middle or realizing the shared random string model. In *43rd FOCS*, pages 345–355. IEEE Computer Society Press, November 2002.

[BF01] Dan Boneh and Matthew Franklin. Identity based encryption from the Weil pairing. Cryptology ePrint Archive, Report 2001/090, 2001. http://eprint.iacr.org/2001/090.

[BGJ+16] Nir Bitansky, Shafi Goldwasser, Abhishek Jain, Omer Paneth, Vinod Vaikuntanathan, and Brent Waters. Time-lock puzzles from randomized encodings. In Madhu Sudan, editor, *ITCS 2016*, pages 345–356. ACM, January 2016.

[BN00] Dan Boneh and Moni Naor. Timed commitments. In Mihir Bellare, editor, *CRYPTO 2000*, volume 1880 of *LNCS*, pages 236–254. Springer, Heidelberg, August 2000.

[BOV05] Boaz Barak, Shien Jin Ong, and Salil Vadhan. Derandomization in cryptography. Cryptology ePrint Archive, Report 2005/365, 2005. http://eprint.iacr.org/2005/365.

[BP04] Boaz Barak and Rafael Pass. On the possibility of one-message weak zero-knowledge. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 121–132. Springer, Heidelberg, February 2004.

[BP15] Nir Bitansky and Omer Paneth. ZAPs and non-interactive witness indistinguishability from indistinguishability obfuscation. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part II*, volume 9015 of *LNCS*, pages 401–427. Springer, Heidelberg, March 2015.

[COSV16a] Michele Ciampi, Rafail Ostrovsky, Luisa Siniscalchi, and Ivan Visconti. 4-round concurrent non-malleable commitments from one-way functions. Cryptology ePrint Archive, Report 2016/621, 2016. http://eprint.iacr.org/2016/621.

[COSV16b] Michele Ciampi, Rafail Ostrovsky, Luisa Siniscalchi, and Ivan Visconti. Concurrent non-malleable commitments (and more) in 3 rounds. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part III*, volume 9816 of *LNCS*, pages 270–299. Springer, Heidelberg, August 2016.

[DDN00] Danny Dolev, Cynthia Dwork, and Moni Naor. Nonmalleable cryptography. *SIAM Journal on Computing*, 30(2):391–437, 2000.

[DN93] Cynthia Dwork and Moni Naor. Pricing via processing or combatting junk mail. In Ernest F. Brickell, editor, *CRYPTO'92*, volume 740 of *LNCS*, pages 139–147. Springer, Heidelberg, August 1993.

[DN00] Cynthia Dwork and Moni Naor. Zaps and their applications. In *41st FOCS*, pages 283–293. IEEE Computer Society Press, November 2000.

[GKS16] Vipul Goyal, Dakshita Khurana, and Amit Sahai. Breaking the three round barrier for non-malleable commitments. In Irit Dinur, editor, *57th FOCS*, pages 21–30. IEEE Computer Society Press, October 2016.

[GL89] O. Goldreich and L. A. Levin. A hard-core predicate for all one-way functions. In *Proceedings of the Twenty-first Annual ACM Symposium on Theory of Computing*, STOC '89, pages 25–32, New York, NY, USA, 1989. ACM.

[GLOV12] Vipul Goyal, Chen-Kuei Lee, Rafail Ostrovsky, and Ivan Visconti. Constructing non-malleable commitments: A black-box approach. In *53rd FOCS*, pages 51–60. IEEE Computer Society Press, October 2012.

[GMPY11] Juan A. Garay, Philip D. MacKenzie, Manoj Prabhakaran, and Ke Yang. Resource fairness and composability of cryptographic protocols. *Journal of Cryptology*, 24(4):615–658, October 2011.

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Non-interactive zaps and new techniques for NIZK. In Cynthia Dwork, editor, *CRYPTO 2006*, volume 4117 of *LNCS*, pages 97–111. Springer, Heidelberg, August 2006.

[Goy11] Vipul Goyal. Constant round non-malleable protocols using one way functions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 695–704. ACM Press, June 2011.

[GPR16] Vipul Goyal, Omkant Pandey, and Silas Richelson. Textbook non-malleable commitments. In Daniel Wichs and Yishay Mansour, editors, *48th ACM STOC*, pages 1128–1141. ACM Press, June 2016.

[GRRV14] Vipul Goyal, Silas Richelson, Alon Rosen, and Margarita Vald. An algebraic approach to non-malleability. In *55th FOCS*, pages 41–50. IEEE Computer Society Press, October 2014.

[JJ99] Markus Jakobsson and Ari Juels. Proofs of work and bread pudding protocols. In *Proceedings of the IFIP TC6/TC11 Joint Working Conference on Secure Information Networks: Communications and Multimedia Security*, CMS '99, pages 258–272, Deventer, The Netherlands, The Netherlands, 1999. Kluwer, B.V.

[Kiy14] Susumu Kiyoshima. Round-efficient black-box construction of composable multi-party computation. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part II*, volume 8617 of *LNCS*, pages 351–368. Springer, Heidelberg, August 2014.

[KS17a] Dakshita Khurana and Amit Sahai. Birthday simulation and applications. 2017.

[KS17b] Dakshita Khurana and Amit Sahai. Two-message non-malleable commitments from standard sub-exponential assumptions. Cryptology ePrint Archive, Report 2017/291, 2017. http://eprint.iacr.org/2017/291.

[LP09] Huijia Lin and Rafael Pass. Non-malleability amplification. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 189–198. ACM Press, May / June 2009.

[LP11] Huijia Lin and Rafael Pass. Constant-round non-malleable commitments from any one-way function. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 705–714. ACM Press, June 2011.

[LPV08] Huijia Lin, Rafael Pass, and Muthuramakrishnan Venkitasubramaniam. Concurrent non-malleable commitments from any one-way function. In Ran Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 571–588. Springer, Heidelberg, March 2008.

[May93] Timothy May. Timed-release crypto. 1993.

[Nak12] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008. 2012.

[Pas13] Rafael Pass. Unprovable security of perfect NIZK and non-interactive non-malleable commitments. In Amit Sahai, editor, *TCC 2013*, volume 7785 of *LNCS*, pages 334–354. Springer, Heidelberg, March 2013.

[PPV08] Omkant Pandey, Rafael Pass, and Vinod Vaikuntanathan. Adaptive one-way functions and applications. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 57–74. Springer, Heidelberg, August 2008.

[PR05a] Rafael Pass and Alon Rosen. Concurrent non-malleable commitments. In *46th FOCS*, pages 563–572. IEEE Computer Society Press, October 2005.

[PR05b] Rafael Pass and Alon Rosen. New and improved constructions of non-malleable cryptographic protocols. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 533–542. ACM Press, May 2005.

[PW10] Rafael Pass and Hoeteck Wee. Constant-round non-malleable commitments from sub-exponential one-way functions. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 638–655. Springer, Heidelberg, May 2010.

[Rog06] Phillip Rogaway. Formalizing human ignorance. In Phong Q. Nguyen, editor, *Progress in Cryptology - VIETCRYPT 06*, volume 4341 of *LNCS*, pages 211–228. Springer, Heidelberg, September 2006.

[RSW96] R. L. Rivest, A. Shamir, and D. A. Wagner. Time-lock puzzles and timed-release crypto. Technical report, Massachusetts Institute of Technology, Cambridge, MA, USA, 1996.

[Wee10] Hoeteck Wee. Black-box, round-efficient secure computation via non-malleability amplification. In *51st FOCS*, pages 531–540. IEEE Computer Society Press, October 2010.