Two-Round Concurrent Non-Malleable Commitment from Time-Lock Puzzles

Huijia Lin* Rafael Pass† Pratik Soni*

March 24, 2017

Abstract

Non-malleable commitment is a fundamental cryptographic tool for preventing man-in-the-middle attacks. Since its proposal by Dolev, Dwork, and Noar in 1991, a rich line of research has steadily reduced the number of communication rounds needed for non-malleable commitment, towards the ultimate goal of constructing non-interactive non-malleable commitment from well-studied hardness assumptions. However, this successful research recently hit a barrier: Pass showed that 2-round non-malleable commitment cannot be based on any, even subexponentially secure, falsifiable assumptions, via black-box security reductions [Pass, STOC 2011], and the only known construction of non-interactive non-malleable commitment is based on a new and non-falsifiable assumption [Pandey, Pass, Vaikuntanathan, Crypto 2008].

In this work, we present a construction of 2-round non-malleable commitment from time-lock puzzles; they are "mechanisms for sending messages to the future" proposed by Rivest, Shamir, and Wagner in 1996, whose original construction has withstood cryptanalysis for two decades. In addition, our construction uses a subexponentially secure injective one-way function and a non-interactive witness indistinguishable proof system. The key to circumventing Pass's impossibility result lies in leveraging the different *nature* of hardness provided by time-lock puzzles and classical cryptographic primitives. Conventionally, cryptographic hardness is defined against adversaries with bounded time (or equivalently circuit-size); in contrast, the hardness of time-lock puzzles holds against adversaries with bounded parallel-time (or circuit-depth). This difference allows us to construct commitment schemes that are simultaneously harder than each other according to different complexity measures, which imply a weak form of non-malleability. It is then strengthened through a new 2-round non-malleability amplification technique, and the final protocol is non-malleable even in the fully concurrent setting.

To the best of our knowledge, this is the first time that time-lock puzzles are used constructively outside time-released cryptography, and opens an interesting direction of combining hardness w.r.t. different complexity measures for achieving cryptographic goals.

*{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 first non-malleable commitment protocol was constructed in the original work of [DDN00] in 1991, based on the minimal assumption of one-way functions. However, their protocol requires $\Omega(\log n)$ rounds of communication, where n is the length of the security parameter. Since then, a central question in the study of non-malleable commitment is determining the exact number of communication rounds needed. 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 non-malleable commitment can be constructed from one-way functions [GRRV14], while 3 round from quasi-polynomially secure injective one-way functions [GPR16, COSV16]. However, the situation changes drastically when it comes to 2 or 1 rounds. First, 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 adaptively secure injective one-way functions, which already has a strong non-malleability flavour. Later, Pass showed that, in fact, 2-round non-malleable commitment cannot be based on any, even subexponentially hard, falsifiable assumptions [Pas11]. His impossibility result is strong and rules out even non-black-box constructions, as long as the security reduction makes only black-box use of the attackers. Nevertheless, we still ask

Can we have 2-round non-malleable commitments from standard assumptions?

2-Round Non-Malleable Commitment Scheme from Time-Lock Puzzles. We make significant progress towards answering the question by constructing a 2-round non-malleable commitment scheme from *time-lock puzzles* — they are mechanisms for "sending messages to the future" proposed fifteen years ago by Rivest, Shamir, and Wagner [RSW96] following May's work on time-release cryptography [May93]. Time-lock puzzles enable 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, the puzzle solution *i)* can be extracted after investing computing power for a sufficiently large amount of time 2^t, but *ii)* is hard to compute for any adversary that runs in *parallel-time* $T = T(t)$ significantly less than 2^t, including resourceful adversaries with a large number of processors whose overall runtime[1] can be $B = B(n)$ significantly larger than 2^t. [2] When representing

[1] The sum of the runtime of all processors running in parallel.

[2] Originally, time-lock puzzles are meant to be solved in polynomial time in order to "send messages to the future", that is, 2^t is polynomial in n. But this is clearly insufficient for building cryptographic primitives that are hard for any polynoimal time attackers. Therefore, in this work, we consider "scaled up" versions of time-lock puzzles that may take time 2^t exponential in n to solve.

parallel adversaries as circuits, this is equivalent to saying that puzzles are hard for adversarial circuits with depth $T(t)$ and size $B(n)$, or $(T(t), B(n))$-hard for short. The "quality" of time-lock puzzles is mainly reflected in the gap between the amount of time 2^t for solving the puzzles, and the parallel time T in which puzzles remain hard. Previous works considered strong puzzles with very small gaps for exponential $T = 2^{\delta t}$ or even strongly exponential $T = \delta 2^t$. In comparison, our construction of 2-round non-malleable commitment only rely on *subexponential* parallel-time hardness, more precisely, hardness against parallel adversaries with subexponential parallel-time $T = 2^{t^\delta}$ and subexponential overall time $B = 2^{n^\varepsilon}$.

Theorem 1 (Main Theorem, Informal)**.** *Let T and B be two arbitrary subexponential functions. There is a construction of 2-round (concurrently) non-malleable commitment scheme from $(T(t), B(n))$-hard time-lock puzzles, assuming the existence of a subexponentially secure injective one-way function and a subexponentially secure non-interactive witness indistinguishable proof system.*

We briefly discuss instantiations of the building blocks of our theorem. First, non-interactive witness indistinguishable (NIWI) proof systems can be constructed from a variety of assumptions [BOV05, GOS06, BP15], in particular from ZAP and a de-randomization assumption [BOV05] (where ZAP in turn can be based on trapdoor permutations [DN00]), or from specific number theoretic assumptions on bilinear pairing groups [GOS06]. For time-lock puzzles, Rivest, Shamir, and Wagner (RSW) [RSW96] proposed a construction based on *repeated squaring modular an RSA-integer*, which is believed to be an *inherently sequential process*: A puzzle has form $(s + g^{2^{2^t}} \bmod N,\ N)$, where $N = pq$ is a randomly chosen n-bit RSA-modulus and g can either be chosen at random or fixed to some particular element (*e.g.*, 2). An honest puzzle generator, knowing the factorization of N, can efficiently generate the puzzle by computing $2^{2^t} \bmod \phi(N)$ before exponentiating. On the other hand, for adversaries that cannot factor N, the best strategy seems to be computing $g^{2^{2^t}} \bmod N$ using 2^t sequential squaring, which reveals s. Twenty years after the original proposal, there is still no strategy that can speed up this computation significantly through parallelization. In fact, current knowledge suggests that the RSW puzzles could be hard for (strongly) exponential parallel-time $T = \delta 2^t$. Our construction relies on much weaker parallel-time hardness as formalized below.

Assumption 1 (Subexponential RSW Assumption)**.** *There exist two subexponential functions T, S and a constant c, such that, for every function t such that $c \log n < t(n) < B(n)$, and every adversary A that runs in $T(t)$-parallel time and $B(n)$-overall time, the probability that $A(N)$ computes $(g^{2^{2^t}} \bmod N)$ is negligible, where N is a randomly chosen n-bit RSA-modulus, and g is chosen at random or fixed appropriately.*

Besides from the RSW puzzles, our non-malleable commitment can also be based on another, more recent, proposal of time-lock puzzles by Bitansky *et al.* [BGJ+16], which can be constructed from any parallel-time hard language, but relies on the existence of indistinguishability obfuscation.

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.

A Perspective and Future Research. The concept of falsifiable assumptions [Nao03] is meant to capture assumptions that can be falsified through a game between an efficient challenger and an alleged attacker with bounded resource, which classically is bounded-time or bounded-size. The hardness of time-lock puzzles can in fact be falsified in the same way, but restricting the attacker to bounded parallel-time or circuit-depth. In turns out that Pass's impossibility result relies on the fact that two or more falsifiable assumptions w.r.t. the same resource can be collapsed into one: Simply have one combined game that plays each game with equal probability. However, assumptions w.r.t. different resources cannot be combined, which allows us to circumvent his impossibility result.

To the best of our knowledge, this is the first time that time-lock puzzles are used constructively for building cryptographic objects outside time-released cryptography. An interesting open problem is whether they can be used in more applications, potentially circumventing other impossibility results. Moreover, beyond the two specific resources — circuit-size and circuit-depth — considered in this work, there are many other resources, for instance, time-hard and memory-hard objects. We believe that understanding how to leverage the synergy of different resources for achieving cryptographic goals an interesting future research direction.

1.1 Organization

In Section 2, we give a detailed overview of our approach for constructing 2-round non-malleable commitments. In Section 3 we detail the preliminaries and definitions, and give a construction for time-lock puzzles. Section 4 gives constructions of basic commitment schemes which are size-robust, depth-robust and size-and-depth robust. Using these basic commitment schemes, we give a construction of a commitment scheme which is non-malleable w.r.t. extraction in Section 5. In Section 6, we describe our non-malleability strengthening technique and detail its proof. Finally in Section 7, we construct 2-round non-malleable commitment scheme for n-bit identities.

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 commitment.

2.1 Towards Overcoming Pass's Impossibility Result

In the literature, there are two formulations of non-malleable commitment, 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 constructed from subexponentially hard falsifiable assumptions, via black-box security reduction. His result directly rules out the feasibility of building 2-round non-malleable commitment from size-robust commitment schemes alone, as they are implied by subexponentially hard one-way functions, and in fact, also applies to using depth-robust commitment alone. However, we show that combining size-robust and depth-robust commitments in non-trivial ways overcomes the impossibility.

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 commitment [3]. 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:

[3] 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.

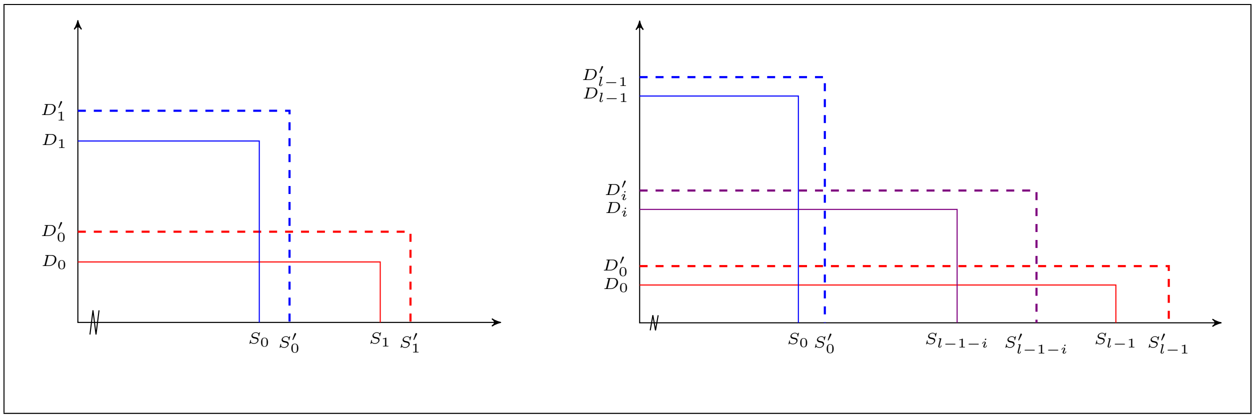

Figure 1: (left) A 1-bit tag based commitment scheme: The tag-0 (resp., tag-1) commitment scheme is hiding for circuits of depth below D_0 (resp., D_1) *OR* size below S_1 (resp., S_0), represented by the solid line joining D_0 (resp., D_1) and S_1 (resp., S_0). The tag-0 (resp., tag-1) commitment scheme 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}'.

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
- 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. [4] 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 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 that circumvents Pass's impossibility result.

- *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

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

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

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 injective one-way functions 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 $0 \le i \le |v|-1$. 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. $d \le S$) if for every $n \in \mathbb{N}$, $d(n) < S(n)$ (resp. $d(n) \le S(n)$). 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 $d \le S << S^*$.

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 $f : \{0,1\}^* \rightarrow \{0,1\}^*$ 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 $L = \{x : \exists w \text{ s.t. } (x,w) \in \mathcal{R}_L\}$.*

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 2. *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 3. *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 $\bot$. 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.

<u>The man-in-the-middle (MIM) execution:</u> 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[5]. In the left interactions, A_n interacts with C and receives commitments to values $v_1, \dots, v_m \in \{0,1\}^n$, using identities $\mathsf{id}_1, \mathsf{id}_2, \dots, \mathsf{id}_m$ of its choice. In the right interactions A_n interacts with R attempting to commit to related values $\tilde{v}_1, \dots, \tilde{v}_m$, using identities $\tilde{\mathsf{id}}_1, \tilde{\mathsf{id}}_2, \dots, \tilde{\mathsf{id}}_m$ of its choice. We define the values $\tilde{v}_i$ committed on the right as $\tilde{v}_i = \mathsf{val}(\tilde{c}_i)$

[5] 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.

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:*

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

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 modulus $N = pq$.
  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.*

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 $\mathsf{poly}(T(t))$. [6] 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. [7]

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

[6] 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 $\delta < \delta' < 1$.

[7] 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 $v[0]|| \dots ||v[n-1]$ as its output.

Theorem 4. *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,
$$\{s \leftarrow \{0,1\}^n,\ Z \leftarrow \mathsf{Gen}(1^n, 1^t, s)\ :\ (s, Z)\}\ , \tag{1}$$
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 5. *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 6. $(\mathsf{ECom}_{d,S}, \mathsf{EOpen}_{d,S})$ *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 $v_0, v_1 \in \{0,1\}^n$,

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

  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 $\mathbb{N}$,

$$n << d_0 << d_1 << \ldots << d_{l-1} << d_l << S_0 << S_1 << \ldots << S_{l-1} << S_l << S^* << 2^{n^{\varepsilon}} \ , \tag{3}$$

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 6 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 7. $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ *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 3). 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,

$$\mathsf{emim}^{A}_{\mathsf{ENMCom}}(v_0)\ ; \mathsf{emim}^{A}_{\mathsf{ENMCom}}(v_1)\ . \tag{4}$$

  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,

$$\begin{aligned}\mathsf{dep}(B) &= \mathsf{dep}(A) + \mathsf{dep}(o\mathcal{E}_{\tilde{\mathsf{id}}}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_0) + \mathsf{poly}(d_{\tilde{\mathsf{id}}+1}) + \mathsf{poly}(n) < \mathsf{poly}(d_{\tilde{\mathsf{id}}+1})\ .\end{aligned} \tag{5}$$

  Similarly, the size of B is such that,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(o\mathcal{E}_{\tilde{\mathsf{id}}}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(S_0) + \mathsf{poly}(S_{l-\tilde{\mathsf{id}}}) + \mathsf{poly}(n)\\ &< \mathsf{poly}(S_{l-\tilde{\mathsf{id}}}) << S^*\ .\end{aligned} \tag{6}$$

We consider two cases for the identities id and $\tilde{\mathsf{id}}$ as follows: [8]

<u>Case 1 - $\mathsf{id} > \tilde{\mathsf{id}}$:</u> 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 $B \in \mathcal{C}^{\vee}_{d_{\mathsf{id}},S_{l-\mathsf{id}-1}}$.

<u>Case 2 - $\mathsf{id} < \tilde{\mathsf{id}}$:</u> 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 which is one-one non-malleable w.r.t. extraction into a 2-round commitment scheme which is concurrent non-malleable w.r.t. extraction as well as concurrent non-malleable (w.r.t. commitment). More precisely, let $\langle C, R\rangle$ be a 2-round tag-based commitment scheme for $t(n)$-bit identities that is non-malleable w.r.t. extraction by an extractor $o\mathcal{E}_{\mathsf{NM}}$ then the transformation results in a 2-round commitment scheme $\langle \widehat{C}, \widehat{R}\rangle$ which is concurrent non-malleable

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

w.r.t. extraction by an extractor $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ and concurrent non-malleable. Given the following hierarchy of non-decreasing functions on $\mathbb{N}$,

$$\begin{aligned} & n << d_2 << d_4 << d_3 << d_1 << d_{\mathsf{NM}} << d'_{\mathsf{NM}} << \\ & S_4 << S_1 << S_{\mathsf{NM}} << S_2 << S_3 << S'_3 << S^* << 2^{n^{\varepsilon}} \ , \end{aligned} \tag{7}$$

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 (d'_{NM}, S_2)-over-extractable by extractor $o\mathcal{E}_{\mathsf{NM}}$. Furthermore, $\langle C, R \rangle$ is one-one $\mathcal{C}^{\wedge}_{d_{\mathsf{NM}}, S_{\mathsf{NM}}}$-non-malleable w.r.t. extraction by $o\mathcal{E}_{\mathsf{NM}}$. [9]

2. $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ is a perfectly binding commitment scheme which is $\mathcal{C}^{\vee}_{d_1, S_1}$-hiding and $(d_{\mathsf{NM}}, S_{\mathsf{NM}})$-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_4, S_3)-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_1, S'_3)-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_1)-over-extractable w.r.t. extractor $o\mathcal{E}_4$.

6. $\mathcal{C}_{S^*}$-secure NIWI, ZAP and OWP f.

Using the above mentioned buiding blocks, the transformation produces $\langle \widehat{C}, \widehat{R} \rangle$ which is a 2-round, tag-based commitment scheme for $t(n)$-bit identities that is $(d_{\mathsf{NM}}, S_{\mathsf{NM}})$-over-extractable w.r.t. an extractor $\widehat{o\mathcal{E}}_{\mathsf{NM}}$. Furthermore, $\langle \widehat{C}, \widehat{R} \rangle$ is both concurrent $\mathcal{C}^{\wedge}_{d_2, S_4}$-non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ and concurrent $\mathcal{C}^{\wedge}_{d_2, S_4}$-non-malleable (w.r.t. commitment). Before we formally describe $\langle \widehat{C}, \widehat{R} \rangle$, we first describe a subprotocol puzz used by the receiver $\widehat{R}$ to generate a puzzle which is sent to the committer $\widehat{C}$ in the first round.

6.1 Subprotocol puzz

A puzzle scheme puzz is a tuple $(\mathsf{Gen}, \mathsf{Validity}, \mathsf{Ver}, \mathsf{Sol})$ with the following syntax:

- **Syntax:**

  1. Gen is a PPT which takes a security parameter 1^n as input and outputs a puzzle Y.
  2. $\mathsf{Validity}$ is a deterministic polynomial time algorithm that checks whether a puzzle Y is valid. We say that Y is a valid puzzle iff $\mathsf{Validity}(1^n, Y) = 1$.
  3. Ver is a deterministic polynomial time computable function which on input puzzle Y and a string s' outputs $0/1$. We say that s' is a solution of a puzzle Y iff $\mathsf{Ver}(1^n, Y, s') = 1$.
  4. $\mathsf{Sol} = \{\mathsf{Sol}_n\}_{n \in \mathbb{N}}$ is a family of non-uniform circuits that for every $n \in \mathbb{N}$ and puzzle Y outputs a string. It guarantees that if the puzzle Y is valid (i.e., $\mathsf{Validity}(1^n, Y) = 1$) then Sol outputs a solution to the puzzle Y with over-whelming probability.

[9] 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.

- **Puzzle construction:** We now describe our construction of a puzzle scheme below. We will be using the commitment scheme $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$, OWP f and NIWI described at the begining of Section 6 as building blocks in the construction.

  – Generation - Algorithm Gen:

    1. On input security parameter 1^n, Gen samples $s_1, s_2 \leftarrow \{0,1\}^n$.
    2. Gen computes commitments c_1 and c_2 to s_1 and s_2 using ECom_4 respectively.
    3. Gen computes $y_1 = f(s_1)$ and $y_2 = f(s_2)$.
    4. Gen uses NIWI to prove that either of the following holds:
       (a) *either* there exists a string $\bar{s_1}$ such that c_1 is a valid commitment to $\bar{s_1}$ and $y_1 = f(\bar{s_1})$,
       (b) *or* there exists a string $\bar{s_2}$ such that c_2 is a valid commitment to $\bar{s_2}$ and $y_2 = f(\bar{s_2})$.

       Let π denote the NIWI proof that proves statement (a) using witness s_1.
    5. Gen returns the tuple $Y = (c_1, c_2, y_1, y_2, \pi)$ as the puzzle.

  – Validation - Algorithm $\mathsf{Validity}$:
    On input security parameter 1^n and a tuple $(c_1, c_2, y_1, y_2, \pi)$, $\mathsf{Validity}$ outputs 1 if the NIWI proof π is accepting. Otherwise, outputs 0.

  – Verification - Function Ver:
    On input security parameter 1^n, a puzzle $(c_1, c_2, y_1, y_2, \pi)$ and a string s', Ver returns 1 if either $y_1 = f(s')$ or $y_2 = f(s')$. Otherwise, it returns 0.

  – Solve - Solver Sol:
    On input security parameter 1^n and a puzzle $(c_1, c_2, y_1, y_2, \pi)$, Sol runs the extractor $o\mathcal{E}_4$ on both c_1 and c_2 obtaining outputs s'_1 and s'_2. If $y_1 = f(s'_1)$, Sol outputs s'_1. Otherwise, Sol outputs s'_2.

Theorem 8. *If* $(\mathsf{ECom}_4, \mathsf{EOpen}_4)$ *is* $\mathcal{C}^{\vee}_{d_4,S_4}$*-hiding and* (d_3, S_1)*-over-extractable w.r.t.* $o\mathcal{E}_4$*, and* NIWI *and* f *are* $\mathcal{C}_{S^*}$*-secure then* $\mathsf{puzz} = (\mathsf{Gen}, \mathsf{Validity}, \mathsf{Ver}, \mathsf{Sol})$ *has the following properties,*

1. *Correctness: There exists a negligible function* $\nu(\cdot)$ *such that for every* $n \in \mathbb{N}$ *and* $Y \in \{0,1\}^{\mathsf{poly}(n)}$,

$$\mathsf{Pr}\left[s' \leftarrow \mathsf{Sol}_n(Y) : \mathsf{Validity}(1^n, Y) = 1 \wedge \mathsf{Ver}(1^n, Y, s') = 0\right] \le \nu(n) \ .$$

2. $\mathcal{C}^{\vee}_{d_4,S_4}$*-hardness: For every non-uniform circuit family* $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\vee}_{d_4,S_4}$*, there exists a negligible function* $\nu(\cdot)$ *such that for every* $n \in \mathbb{N}$,

$$\mathsf{Pr}\left[Y \leftarrow \mathsf{Gen}(1^n), s' \leftarrow A_n(Y) : \mathsf{Ver}(1^n, Y, s') = 1\right] \le \nu(n) \ .$$

3. $\mathcal{C}^{\wedge}_{d_3,S_1}$*-solvable:* Sol *belongs to the circuit class* $\mathcal{C}^{\wedge}_{d_3,S_1}$.

Proof. We discuss all the properties in the following:

1. For any $Y = (c_1, c_2, y_1, y_2, \pi) \in \{0,1\}^{\mathsf{poly}(n)}$ if $\mathsf{Validity}(1^n, Y) = 1$ then due to the soundness of NIWI either c_1 is a commitment to s_1 and $y_1 = f(s_1)$ or c_2 is a commitment to s_2 and $y_2 = f(s_2)$. Therefore, at least one of c_1 or c_2 is a valid commitment for ECom_4. Since,

Sol runs the extractor on both commitments c_1 and c_2, we can w.l.o.g. assume that c_1 is a valid commitment to s_1 and $y_1 = f(s_1)$. Since, Sol runs the extractor $o\mathcal{E}_4$ on c_1, it extracts out the correct committed value s_1 (except with negligible probability) due to the over-extractability of ECom_4. Moreover, Ver accepts the extracted value as the solution to Y, that is, $\mathsf{Ver}(1^n, Y, s_1) = 1$. Hence, if $\mathsf{Validity}(1^n, Y) = 1$, Sol outputs a solution except with negligible probability.

2. Let us assume for contradiction that there exists a family of non-uniform circuits $A = \{A_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\vee}_{d_4,S_4}$ and a polynomial $p(\cdot)$ such that for infinitely many $n \in \mathbb{N}$, A solves honestly generated puzzles with probability $\frac{1}{p(n)}$; that is,

$$\mathsf{Pr}\left[Y \leftarrow \mathsf{Gen}(1^n), s' \leftarrow A_n(Y) : \mathsf{Ver}(1^n, Y, s') = 1\right] \geq \frac{1}{p(n)} \ . \tag{8}$$

By our construction of puzz, an honestly generated puzzle Y is the tuple $(c_1, c_2, y_1, y_2, \pi)$ where c_i is a commitment to a random n-bit string s_i and $y_i = f(s_i)$ for $i \in \{1, 2\}$. The proof π proves that c_1 is a commitment to s_1 and $y_1 = f(s_1)$. Furthermore, by the definition of Ver, Y has two solutions, namely s_1 and s_2. Therefore, we have,

$$\mathsf{Pr}\left[Y \leftarrow \mathsf{Gen}(1^n), Y = (c_1, c_2, y_1, y_2, \pi), s' \leftarrow A_n(Y) : s' \in \{s_1, s_2\}\right] \geq \frac{1}{p(n)} \ . \tag{9}$$

Then, A outputs at least one of s_1 or s_2 with probability at least $\frac{1}{2p(n)}$. W.l.o.g, we assume that it outputs s_1; that is,

$$\mathsf{Pr}\left[Y \leftarrow \mathsf{Gen}(1^n), Y = (c_1, c_2, y_1, y_2, \pi), s' \leftarrow A_n(Y) : s' = s_1\right] \geq \frac{1}{2p(n)} \ . \tag{10}$$

Using A, we construct a non-uniform circuit family $B = \{B_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\vee}_{d_4,S_4}$ that inverts the OWP f with non-negligible probability $\frac{1}{3p(n)}$. B on receiving the OWP challenge $y_1 = f(s_1)$ samples a random n-bit string s_2. It then computes a commitment c_1 to 0^n and a commitment c_2 to s_2 using ECom_4. Furthermore, it assigns $y_2 = f(s_2)$ and computes a NIWI proof to prove that c_2 is a commitment to s_2 and $y_2 = f(s_2)$ (i.e., statement (b) in Step 4 of Gen). It then internally runs A with $(c_1, c_2, y_1, y_2, \pi)$ and forwards the output of A as its output. For B to be able to invert OWP with probability $\frac{1}{3p(n)}$, A must output s_1 with probability at least $\frac{1}{3p(n)}$ on input Y. Note that the distribution of Y generated by B is different from the distribution of puzzles generated by $\mathsf{Gen}(1^n)$. Therefore, we need to show that A continues to do well on this new distribution of Y.

We proceed to prove that A on input Y, sampled from the distribution due to B, outputs s_1 with probability $\frac{1}{3p(n)}$ via a sequence of hybrid distributions of Y.

Hybrid H_0 : This hybrid samples Y honestly by running $\mathsf{Gen}(1^n)$. By Equation (10), the probability that A on input Y outputs s_1 is at least $\frac{1}{2p(n)}$.

Hybrid H_1 : This hybrid samples Y identically to Hybrid H_0 except that the proof π in Y is generated differently. Instead of generating π that proves c_1 is a valid commitment to s_1 and $y_1 = f(s_1)$, H_1 generates π that proves c_2 is a valid commitment to s_2 and $y_2 = f(s_2)$. The only difference between hybrids H_1 and H_0 is the witness used to generate the proof π. Note that any $A \in \mathcal{C}^{\vee}_{d_4,S_4}$ has size at most $\mathsf{poly}(S^*)$. Since, NIWI is $\mathcal{C}_{S^*}$-witness-indistinguishable, the probability that A on input $(c_1, c_2, y_1, y_2, \pi)$ outputs s_1 is then at least $\frac{1}{2p(n)} - negl$.

Hybrid H_2 **:** This hybrid samples Y identically to H_1, except that c_1 in Y is generated differently. In H_1, c_1 is a commitment to s_1 such that $y_1 = f(s_1)$. In this hybrid, c_1 is a commitment to 0^n but $y_1 = f(s_1)$. The rest of the sampling in H_2 is the same as that in H_1, that is, c_2 is a commitment to s_2, $y_2 = f(s_2)$ and the proof π proves that c_2 is a commitment to s_2 and $y_2 = f(s_2)$. The only difference between H_2 and H_1 is the commitment c_1 which in H_1 commits to s_1 but in H_2 commits to 0^n. One can show that due to the $\mathcal{C}^{\vee}_{d_4,S_4}$-hiding of ECom_4, the probability that A on input $(c_1, c_2, y_1, y_2, \pi)$ outputs s_1 is at least $\frac{1}{2p(n)} - negl$. Infact, note that the distribution of Y sent to A in H_2 is identical to the distribution of Y generated by B on input the OWP challenge y_1. Therefore, A on input Y (generated by B) outputs s_1 with probability at least $\frac{1}{2p(n)} - negl \geq \frac{1}{3p(n)}$. Thus, B inverts OWP f with probability $\frac{1}{3p(n)}$, which is non-negligible.

Finally, note that $B \in \mathcal{C}^{\vee}_{d_4,S_4}$, since $A \in \mathcal{C}^{\vee}_{d_4,S_4}$ and rest of the computation (generating Y) takes $\mathsf{poly}(n)$ time. Since the maximum size of any such B is bounded by $\mathsf{poly}(S^*)$, B violates the $\mathcal{C}_{S^*}$-security of OWP, which gives a contradiction.

3. Sol on input a valid puzzle $Y = (c_1, c_2, y_1, y_2, \pi)$ runs the extractor $o\mathcal{E}_4$ on both c_1 and c_2. Since $o\mathcal{E}_4 \in \mathcal{C}^{\wedge}_{d_3,S_1}$ and rest of the computation done by Sol (checking the consistency of the extracted values with OWP images) takes $\mathsf{poly}(n)$ time, we have $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$.

□

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

We now describe our transformation from a commitment scheme $\langle C, R \rangle$ that is one-one non-malleable w.r.t. extraction to a commitment scheme $\langle \widehat{C}, \widehat{R} \rangle$ that is concurrent non-malleable w.r.t. extraction and also w.r.t. commitment. Our transformation uses the puzz scheme constructed in Section 6.1 and four commitment schemes $\{\mathsf{ECom}_i, \mathsf{EOpen}_i\}_{i \in \{1,2,3,4\}}$ and ZAP as described at the beginning of Section 6.

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 puzzle $Y \leftarrow \mathsf{Gen}(1^n)$.
  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 $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ as the first round message to $\widehat{C}$.

- Commit stage - Second round:

  1. $\widehat{C}$ checks if the puzzle Y is valid and aborts if $\mathsf{Validity}(1^n, Y) = 0$.
  2. (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.
  3. (a) $\widehat{C}$ computes a commitment $c2$ to a random n-bit 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 $r2$.

4. 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 $\bar{s}$ such that $c2$ is a commitment to $\bar{s}$ and $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commit to a decommitment of $c2$ and $\mathsf{Ver}(1^n, Y, \bar{s}) = 1$.

   $\widehat{C}$ proves the statement (a) by using the witness $(v, d1)$.

5. $\widehat{C}$ sends $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ 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 c_1 (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, statistically binding, $\mathcal{C}^{\wedge}_{d_2,S_4}$-hiding, $(d_{\mathsf{NM}}, S_{\mathsf{NM}})$-over-extractable commitment scheme for identities of length $t(n)$.*

Proof. The statistical binding follows from the binding of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$. 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 $\mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$, since $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$ 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 $(d_{\mathsf{NM}}, S_{\mathsf{NM}})$-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 $\mathcal{C}^{\wedge}_{d_2,S_4}$-non-malleable w.r.t. extraction by $\widehat{o\mathcal{E}_{\mathsf{NM}}}$.*

Theorem 11. *$\langle\widehat{C}, \widehat{R}\rangle$ is concurrent $\mathcal{C}^{\wedge}_{d_2,S_4}$-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 $\tilde{A}$ and a distinguisher $\tilde{D}$ that violate one-many non-malleability. Their reduction blows up the size and the depth of the adversary $\tilde{A}$ and the distinguisher $\tilde{D}$ (over A and D respectively) by a $\mathsf{poly}(n)$ factor and thereby inccurs 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 $\mathcal{P}/\text{poly}$, 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 $\mathcal{P}/\text{poly}$.*

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 $\mathcal{C}^{\wedge}_{d_2,S_4}$ 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 show the following: for any fixed $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$ participating in one-left interaction and $m = \mathsf{poly}(n)$ many right interactions and any fixed sequences $\{v_0\}_{n\in\mathbb{N}}$ and $\{v_1\}_{n\in\mathbb{N}}$, the following computational indistinguishability holds,

a)
$$\mathsf{emim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_0) \approx_c \mathsf{emim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_1)\ , \tag{11}$$

where the random variable $\mathsf{emim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v)$ describes the view of A and values extracted from right interactions by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ when the value committed in the left interaction is v.

b)
$$\mathsf{mim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_0) \approx_c \mathsf{mim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v_1)\ , \tag{12}$$

where the random variable $\mathsf{mim}^{A}_{\langle \widehat{C},\widehat{R}\rangle}(v)$ describes the view of A and values A commits to in the right interactions when the value committed in the left interaction is v.

We prove the indistinguishability of the above pairs of random variables (Equations (11), (12)) via a sequence of hybrids $\{H_j(v)\}_{j\in[6]}$ for $v \in \{v_0, v_1\}$. Each hybrid $H_j(v)$ runs a MIM execution with A and commits to the value v in the left interaction. We will use the following notation throughout our proof: By $\tilde{x}_i$ we denote a value associated with the ith right interaction. For example, $\tilde{Y}_i$ is the puzzle sent by the honest receiver $\widehat{R}$ to A in the ith right interaction. For each hybrid $H_j(v)$, we denote by $\mathsf{emim}^A_{H_j}(v)$, the random variable that describes the view of A and the values $\{\tilde{v}'_i\}_{i\in[m]}$ extracted from commitments $\{\tilde{c}_i\}_{i\in[m]}$, where $\tilde{c}_i$ is the commitment in the ith right interaction. Similarly, we denote by $\mathsf{mim}^A_{H_j}(v)$, the random variable that describes the view of A in $H_j(v)$ and the values $\{\tilde{v}_i\}_{i\in[m]}$ that A commits to in right interactions. Recall that, if for any right interaction i, the identity $\tilde{\mathsf{id}}_i$ equals the left identity id then we set $\tilde{v}'_i = \perp$ (resp., $\tilde{v}_i = \perp$). Moreover, for notational convenience, we will say that a right interaction i is *successful* if it is accepting and the identity $\tilde{\mathsf{id}}_i$ is different from the left interaction identity id.

For $v \in \{v_0, v_1\}$ and each pair of neighbouring hybrids H_j and H_{j+1}, we show that the view of A along with the values extracted from right interactions are indistinguishable. That is,

Lemma 2. *For $v \in \{v_0, v_1\}$ and $j \in [5]$, 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* $\mathsf{emim}^A_{H_5}(v) = \mathsf{emim}^A_{H_4}(v_0)$.

Furthermore, we show that in each of the hybrids $H_j(v)$ the view of A and the values extracted from right interactions are statistically close to the view of A and the values A commits to in the right interactions. That is,

Lemma 3. *For $v \in \{v_0, v_1\}$ and $j \in [6]$, the following are statistically close,*

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

Therefore, if Lemma 2 holds for all adjacent hybrids and Lemma 3 holds for all hybrids $H_j(v)$ then it is clear that Equation (11),(12) hold.

Recall that, the honest committer $\widehat{C}$ (while committing to a value v) computes a ZAP proof for an OR-statement (see Step 4 of $\widehat{C}$ in $\langle \widehat{C}, \widehat{R}\rangle$). For a commitment c generated by any committer, to be accepting, the committer must prove at least one of statement (a) or (b) as described in Step 4, with an appropriate witness w. Otherwise, with over-whelming probabiltity the ZAP proof $(a_{\mathsf{ZAP}}, b_{\mathsf{ZAP}})$ will not verify and hence the commitment c will not be accepting.

We refer to a witness $w = (v, d1)$ used to prove statement (a) — $c3$ commits to (v, d_1) which is a decommitment of $c1$ — as a *honest* witness. Note that, the honest committer $\widehat{C}$ uses a honest witness to compute the ZAP proof. Similarly, we refer to the the witness (s, d) used to prove the statement (b) — $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commit to (s, d) which is a decommitment of $c2$ to a solution s of the puzzle Y — as a *fake* witness. We say that A (acting as a committer) commits to a fake witness in a right interaction i, if the pair $(\tilde{s}_i, \tilde{d}_i)$ committed by A under $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness — a decommitment of $\tilde{c2}_i$ to a solution $\tilde{s}_i$ of the puzzle $\tilde{Y}_i$. Furthermore, for A to be able to use the fake witness to compute the ZAP proof in an interaction, it must commit to it using $\langle C, R\rangle$.

No-fake-witness invariant. We show that in all hybrids H_j it is only with negligible probability that A commits to a fake witness in any successful interaction. We refer to this condition as the *No-fake-witness* invariant (defined below). We show that this invariant holds in all hybrids $H_j(v)$ for $v \in \{v_0, v_1\}$.

Invariant 1 (No-fake-witness invariant)**.** *For all right interactions i in hybrid $H_j(v)$, the probability that i is successful and A commits to a fake witness in the interaction, is negligible.*

Let us, for now, assume that Invariant 1 holds for $v \in \{v_0, v_1\}$ and hybrid $H_j(v)$. Furthermore, let us consider the values $\tilde{v}_i'$ (extracted) and $\tilde{v}_i$ (committed) described by random variables emim and mim respectively, for some right interaction i. If the interaction i is not successful, that is, either the ZAP proof $(\tilde{a}_{\mathsf{ZAP}i}, \tilde{b}_{\mathsf{ZAP}i})$ is not accepting or the identity $\tilde{\mathsf{id}}_i$ equals the left identity id, then we know that $\tilde{v}_i' = \tilde{v}_i = \bot$. When the right interaction i is successful, $\tilde{v}_i'$ is the value extracted by $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ from $\tilde{c}_i$ (commitment in the ith interaction). For a successful right interaction the ZAP proof is accepting, then by the construction of $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ we know that $\tilde{v}_i'$ is the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$. Since we assumed that Invariant 1 holds, A does not commit to a fake witness in this (successful) interaction. Then, by the soundness of ZAP, A must have used the honest witness to compute the accepting ZAP proof $(\tilde{a}_{\mathsf{ZAP}i}, \tilde{b}_{\mathsf{ZAP}i})$. That is, A must have proved that $\tilde{c1}_i$ is a valid commitment for ECom_1. By the over-extractability of ECom_1 w.r.t. $o\mathcal{E}_1$, $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ extracts $\mathsf{val}(\tilde{c}_i)$ from $\tilde{c}_i$, except with negligible probability. That is, both $\tilde{v}_i'$ and $\tilde{v}_i$ are identical except with negligible probability. Therefore, if Invariant 1 holds for $v \in \{v_0, v_1\}$ then the extracted values $\tilde{v}_i'$ are identical to the committed values $\tilde{v}_i$. Hence, $\mathsf{emim}^A_{H_j}(v)$ is identical to $\mathsf{mim}^A_{H_j}(v)$, except with negligible probability (i.e., Lemma 3 holds).

However, instead of proving about A committing to a fake witness, we prove that the value $(\tilde{s}_i', \tilde{d}_i')$ *extracted* from $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is not a fake witness. That is, for a right interaction i the probability that it is successful and the extracted value $(\tilde{s}_i', \tilde{d}_i')$ satisfies the following, is negligible.

$$\mathsf{Ver}(1^n, \tilde{Y}_i, \tilde{s_i}') = 1 \wedge \mathsf{EOpen}_2(\tilde{c_{2i}}, \tilde{s_i}', \tilde{d_i}') \ .$$

This is captured in the following,

Invariant 2. *For all right interactions i in hybrid $H_j(v)$, the probability that i is successful and the value extracted from $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

We would like to argue that if Invariant 2 holds in $H_j(v)$ then so does Invariant 1. Let us assume for contradiction that Invariant 2 holds and Invariant 1 fails to hold. Then, there exists a right interaction k for which the probability that it is successful and the value committed under $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$ is a fake witness is $1/p(n)$, for some polynomial $p(n)$. By the over-extractability of $\langle C, R\rangle$ w.r.t. extractor $o\mathcal{E}_{\mathsf{NM}}$, the value extracted from $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$ is identical to the value committed, except with negligible probability. Therefore, for this right interaction k, the probability that it is successful and the value extracted from $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness is at least $\frac{1}{2p(n)}$, which is non-negligible. This contradicts that Invariant 2 holds.

By the above observation we can infer that Invariant 2 implies Lemma 3. This implication is captured in the following,

Lemma 4. *For $v \in \{v_0, v_1\}$ and $j \in [6]$, if for every right interaction i in hybrid $H_j(v)$, the probability that interaction i is successful and the value extracted from $(a_{\tilde{\mathsf{NM}}i}, b_{\tilde{\mathsf{NM}}i})$ is a fake witness, is negligible, then the following are statistically close,*

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

Finally, we claim that to conclude the proofs of Theorem 10 and Theorem 11, it is sufficient to prove that Lemma 2 holds for all adjacent hybrids and Invariant 2 holds for all hyrbids. Next, we describe our hybrids $\{H_j(v)\}_{j\in[6]}$ 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 $H_0(v)$.

Claim 1. *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 $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. To show that Claim 1 holds, it suffices to show that for every right interaction i, the probability that it is successful and the value $\tilde{s}'_i$ extracted from $\tilde{c2}_i$ (the commitment using ECom_2) is a solution to the puzzle $\tilde{Y}_i$, is negligible. This is because for the value extracted from $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$ to be a fake witness in a successful interaction i, it must hold that $\tilde{c2}_i$ is committing to a solution to the puzzle $\tilde{Y}_i$. Therefore, if the value extracted from $\tilde{c2}_i$ is not a solution then the value committed under $\tilde{c2}_i$ is not a solution, except with negligible probability. This is because of the over-extractability of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ which guarantees that the value committed under $\tilde{c2}_i$ is identical to the value extracted from it, except with negligible probability.

Now, assume for contradicton that there exists $v \in \{v_0, v_1\}$, a polynomial p and a right interaction k such that k is successful and the value $\tilde{s}'_k$, extracted from $\tilde{c2}_k$, is a solution of the puzzle $\tilde{Y}_k$ with probability at least $1/p(n)$. Then, using A, we construct a non-uniform circuit $B = \{B_n\}_{n\in\mathbb{N}} \in \mathcal{C}^{\vee}_{d_4,S_4}$ that inverts honestly generated puzzles with probability at least $1/p(n)$. More concretely, B with v and k hard-wired in it, on receiving an honestly generated puzzle Y^*, emulates $H_0(v)$ for A except for the kth right interaction. In the kth right interaction, B honestly computes the first message $\tilde{a}_{\mathsf{NM}k}$ of $\langle C, R\rangle$ and the first message $\tilde{a}_{\mathsf{ZAP}k}$ of ZAP (as in $H_0(v)$) and sends the tuple $(\tilde{Y}_k = Y^*, \tilde{a}_{\mathsf{ZAP}k}, \tilde{a}_{\mathsf{NM}k})$ 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 $\tilde{c2}_k$ to extract the value $\tilde{s}'_k$ and returns it as its output (irrespective of whether k is successful or not). Note that B perfectly emulates $H_0(v)$ for A as the distribution of the puzzle received by B is identical to the distribution of the puzzle sent by the honest receiver $\widehat{R}$ of $\langle \widehat{C}, \widehat{R}\rangle$. Then by our hypothesis, $\tilde{s}'_k$ is the solution of the puzzle $\tilde{Y}_k = Y^*$ with probability at least $1/p(n)$.

Furthermore, we argue that B belongs to the circuit class $\mathcal{C}^{\vee}_{d_4,S_4}$: 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 $o\mathcal{E}_2 \in \mathcal{C}^{\wedge}_{d_4,S_3}$ and $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$ we have,

$$\begin{aligned}\mathsf{dep}(B) &= \mathsf{dep}(A) + \mathsf{dep}(o\mathcal{E}_2) + \mathsf{poly}(n)\\ &\le \mathsf{poly}(d_2) + \mathsf{poly}(d_4)\\ &< \mathsf{poly}(d_4) \qquad \text{(since, } d_4 >> d_2 \text{ from Equation (7))}\end{aligned}$$

and $\mathsf{size}(B) < \mathsf{poly}(S^*)$. Therefore, B belongs to the class $\mathcal{C}_{d_4}$ which contradicts the $\mathcal{C}^{\vee}_{d_4,S_4}$-hardness of puzz. □

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 solution s to the puzzle Y. More precisely, $H_1(v)$ first runs $\mathsf{Sol}(Y)$ to obtain a solution s and then commits to s using ECom_2. By Theorem 8, we know that Sol succeeds in computing a solution for a valid Y with over-whelming probability. 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 solution s of the puzzle Y.

First, we show that Invariant 2 holds in $H_1(v)$.

Claim 2. *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 $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists $v \in \{v_0, v_1\}$, a polynomial p and a right interaction k such that k is successful and the value $(\tilde{s}'_k, \tilde{d}'_k)$, extracted from $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$, is a fake witness with probability at least $1/p(n)$. Then, using A we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_2,S_2}$ that violates the hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ with advantage at least $1/2p(n)$.

The circuit B with v and k hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ and internally emulates an execution of $H_1(v)$ with A as follows: [10]

- Step 1: On receiving the first message $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a solution s to the puzzle Y by running Sol.

- Step 2: It samples a random string $r1$, and in the hiding game of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ it sends r_1 and s 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 $(c1, c3, b_{\mathsf{NM}})$ 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 $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ 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 $o\mathcal{E}_{\mathsf{NM}}$ on $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$ and outputs 1 iff the extracted value $(\tilde{s}'_k, \tilde{d}'_k)$ is a fake witness (i.e., B outputs 1 iff $\tilde{s}'_k$ is a solution of the puzzle $\tilde{Y}_k$ and $\mathsf{EOpen}_2(\tilde{c2}_k, \tilde{s}'_k, \tilde{d}'_k) = 1$).

It is easy to see that if B receives a commitment to the random string $r1$, then it perfectly emulates $H_0(v)$ for A and if it receives a commitment to the solution s of the puzzle Y then it perfectly emulates $H_1(v)$ for A. By Claim 1, 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 hiding of $(\mathsf{ECom}_2, \mathsf{EOpen}_2)$.

Moreover, we show that $B \in \mathcal{C}^{\vee}_{d_2,S_2}$: B internally runs $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$, $o\mathcal{E}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{d'_{\mathsf{NM}},S_2}$, and the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}\mathsf{dep}(B) \leq \mathsf{size}(B) &= \mathsf{size}(A) + +\mathsf{size}(\mathsf{Sol}) + \mathsf{size}(o\mathcal{E}_{\mathsf{NM}}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(S_2)\\ &< \mathsf{poly}(S_2) \qquad \text{(since, } S_2 >> S_1, S_4 \text{ from Equation (7))}\end{aligned}$$

Therefore, B belongs to the circuit class $\mathcal{C}_{S_2}$ (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. □

Claim 3. *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 $v \in \{v_0, v_1\}$, a distinguisher $D \in \mathcal{P}/\mathrm{poly}$ and a polynomial p such that D distinguishes $\mathsf{emim}^{A}_{H_0}(v)$ from $\mathsf{emim}^{A}_{H_1}(v)$ with probability $\frac{1}{p(n)}$. Then using A and D, we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_2,S_2}$ that violates the hiding of

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

$(\mathsf{ECom}_2, \mathsf{EOpen}_2)$ with non-negligible advantage $\frac{1}{p(n)}$. B is similar in spirit to the circuit described in the proof of Claim 2.

B with v and k hard-wired in it, participates in the hiding game of ECom_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 2.
- Step 4: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\tilde{c1}_i$ to obtain values $\tilde{v}'_i$. For every unsuccessful right interaction i, B sets $\tilde{v}'_i = \bot$.
- Step 5: B then runs D with the view of A and the values $\{\tilde{v_i}'\}_{i\in[m]}$ 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 $H_0(v)$ for A and if it receives a commitment to the solution s of the puzzle Y then it perfectly emulates $H_1(v)$ for A. Moreover, for every successful interaction i, B sets $\tilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$ and for every unsuccessful interaction, it sets $\tilde{v}'_i = \bot$. 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 $1/p(n)$, B wins the hiding game with advantage at least $1/p(n)$.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_2,S_2}$: Apart from running A and Sol, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\tilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$, we have,

$$\begin{aligned}\mathsf{dep}(B) \leq \mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(\mathsf{Sol}) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_{\mathsf{NM}})\\ &< \mathsf{poly}(S_2) \qquad \text{(since, } S_2 >> S_4, S_{\mathsf{NM}}, S_1 \text{ from Equation (7))}\end{aligned}$$

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 second message b_{NM} of $\langle C, R\rangle$ sent to A in the left interaction is generated differently. In $H_1(v)$, b_{NM} is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to a random string $r2$ whereas in $H_2(v)$ b_{NM} is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commit to a decommitment of $c2$ to a solution to the puzzle Y. More precisely, $H_2(v)$ generates a commitment $c2$ to the solution s (obtained by running $\mathsf{Sol}(Y)$). Let d be the corresponding decommitment string. Then, given a_{NM}, $H_2(v)$ computes the second message b_{NM} to commit to (s, d). 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 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, d).

First, we show that Invariant 2 holds in $H_2(v)$.

Claim 4. *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 $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists $v \in \{v_0, v_1\}$, a polynomial p and a right interaction k such that k is successful and the value $(\tilde{s}'_k, \tilde{d}'_k)$, extracted from $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$, is a fake witness with probability at least $1/p(n)$. Then, using A we construct a non-uniform circuit

$A_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$, 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 $H_2(v)$ 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 $\tilde{\mathsf{id}}_k$ be the respective identities.

- Step 2: A_{NM} selects identity $\mathsf{id}_l = \mathsf{id}$ for its left interaction and identity $\mathsf{id}_r = \tilde{\mathsf{id}}_k$ for its right interaction r. On receiving the first-round message $a_{\mathsf{NM}r}$ from R, A_{NM} samples a puzzle $\tilde{Y}_k$ and the first message of ZAP, $\tilde{a}_{\mathsf{ZAP}k}$. It sends the tuple $(\tilde{Y}_k, \tilde{a}_{\mathsf{NM}k} = a_{\mathsf{NM}r}, \tilde{a}_{\mathsf{ZAP}k})$ as the first-round message to A in the kth right interaction.

- Step 3: On receiving the first message $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, A_{NM} obtains a solution s to the puzzle Y by running Sol.

- Step 4: A_{NM} computes commitments $(c1, c2, c3)$ honestly (as in $H_2(v)$). Let d be the decommitment string of the commitment $c2$, which commits to the solution s.

- 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, d) 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, d).

- Step 6: A_{NM} computes the second message of ZAP (b_{ZAP}) by setting $b_{\mathsf{NM}} = b_{\mathsf{NM}l}$. Then, it sends $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as the second round message to A in the left interaction.

- Step 7: On receiving the second message $(\tilde{c1}_k, \tilde{c2}_k, \tilde{c3}_k, \tilde{b}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{ZAP}k})$ from A in the kth right interaction, B forwards $b_{\mathsf{NM}r} = \tilde{b}_{\mathsf{NM}k}$ 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 $(\tilde{a}_{\mathsf{ZAP}k}, \tilde{b}_{\mathsf{ZAP}k})$ 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 $\tilde{c2}_k$ to a solution of the puzzle $\tilde{Y}_k$.

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 $H_1(v)$ for A and if $b_{\mathsf{NM}l}$ is such that $(a_{\mathsf{NM}l}, b_{\mathsf{NM}l})$ commit to (s, d) then it perfectly emulates $H_2(v)$ for A. By Claim 2, 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 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 $A_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$ and $D_{\mathsf{NM}} \in \mathcal{P}/\text{poly}$: Firstly, it is easy to see that $D_{\mathsf{NM}} \in \mathcal{P}/\text{poly}$ as all the computation done by D_{NM} only takes polynomial time. Next, for A_{NM}:

A_{NM} internally runs $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$, and the rest of the computation done by A_{NM} takes $\mathsf{poly}(n)$ time. Therefore, the depth $\mathsf{dep}(A_{\mathsf{NM}})$ and size $\mathsf{size}(A_{\mathsf{NM}})$ of A_{NM} satisfies the following,

$$\begin{aligned}\mathsf{dep}(A_{\mathsf{NM}}) &= \mathsf{dep}(A) + \mathsf{dep}(\mathsf{Sol}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(d_2) + \mathsf{poly}(d_3)\\ &< \mathsf{poly}(d_{\mathsf{NM}}) \qquad \text{(since, } d_{\mathsf{NM}} >> d_2, d_3 \text{ from Equation (7))}\end{aligned}$$

$$\begin{aligned}\mathsf{size}(A_{\mathsf{NM}}) &= \mathsf{size}(A) + \mathsf{size}(\mathsf{Sol}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(S_4) + \mathsf{poly}(S_1)\\ &< \mathsf{poly}(S_{\mathsf{NM}}) \qquad \text{(since, } S_{\mathsf{NM}} >> S_4, S_1 \text{ from Equation (7))}\end{aligned} \tag{13}$$

Therefore, A_{NM} belongs to the circuit class $\mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$ which contradicts the $\mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$-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,d) along with the first message a_{NM}. The committer C is expected to commit to either $r2$ or (s,d). Note that the challenges $r2$ and (s,d) 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 ones considered in the non-malleability definitions making the above reduction void. However, this issue can be fixed 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 [COSV16] where instead of committing to the preimage s of the OWP challenge Y under the non-malleable commitment scheme, their protocol commits to a random share s_0 of the solution using the non-malleable commitment scheme and sends the other share s_1 in the clear to the receiver. This allows the challenge messages to be fixed before the execution. Note that a similar approach can be adopted in our case. However, for ease of explanation we avoid the fix in our protocol.*

Claim 5. *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 $v \in \{v_0, v_1\}$, a distinguisher $D \in \mathcal{P}/\text{poly}$ and a polynomial p 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 $B \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$ 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 4.

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 $H_2(v)$ with A as follows:

- Step 1: On receiving the first message $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a solution s to the puzzle Y by running Sol.

- Step 2: B computes commitments $(c1, c2, c3)$ honestly (as in $H_2(v)$). Let d be the decommitment string of the commitment $c2$, which commits to the solution s.

- 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, d) as challenges and receives the second message b_{NM} such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ either commit to $r2$ or (s, d).

- Step 4: A_{NM} computes the ZAP proof and sends $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ 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 $\tilde{c1}_i$ to extract values $\tilde{v}'_i$. For every unsuccessful right interaction i, B sets $\tilde{v}'_i = \perp$.

- Step 6: B then runs D with the view of A and the values $\{\tilde{v}_i{}'\}_{i\in[m]}$ 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 $H_1(v)$ for A and if b_{NM} is such that $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to (s, d), then it perfectly emulating $H_2(v)$ for A. Moreover, for every successful interaction i, B sets $\tilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$ and for every unsuccessful interaction B sets $\tilde{v}'_i = \perp$. 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 $B \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$: Apart from running A and Sol, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\tilde{c1}_i$, and the rest of the computation takes polynomial time (including running D). Since, $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$, the depth $\mathsf{dep}(A_{\mathsf{NM}})$ and size $\mathsf{size}(A_{\mathsf{NM}})$ of A_{NM} satisfies the following,

$$\begin{aligned}
\mathsf{dep}(A_{\mathsf{NM}}) &= \mathsf{dep}(A) + \mathsf{dep}(\mathsf{Sol}) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(d_2) + \mathsf{poly}(d_3) + \mathsf{poly}(n) \cdot \mathsf{poly}(d_{\mathsf{NM}}) \\
&< \mathsf{poly}(d_{\mathsf{NM}}) \qquad \text{(since, } d_{\mathsf{NM}} >> d_2, d_3 \text{ from Equation (7))}
\end{aligned}$$

$$\begin{aligned}
\mathsf{size}(A_{\mathsf{NM}}) &= \mathsf{size}(A) + \mathsf{size}(\mathsf{Sol}) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_{\mathsf{NM}}) \qquad (14) \\
&< \mathsf{poly}(S_{\mathsf{NM}}) \qquad \text{(since, } S_{\mathsf{NM}} >> S_4, S_1 \text{ from Equation (7))}
\end{aligned}$$

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

Hybrid $H_3(v)$ **:** Hybrid $H_3(v)$ proceeds identically to $H_2(v)$ except that the second message b_{ZAP} of ZAP sent to A in the left interaction is generated differently. In $H_2(v)$, b_{ZAP} is computed using the witness $(v, d1)$ which is the decommitment of the commitment $c1$ whereas in $H_3(v)$ b_{ZAP} is computed using the witness (s, d) which is the decommitment of $c2$ to a solution of the puzzle Y. More precisely, $H_3(v)$ computes $(c1, c2, c3, b_{\mathsf{NM}})$ identical to $H_2(v)$. Using the witness (s, d) and the first message a_{NM}, B runs the ZAP prover to computes the second message b_{ZAP} where (s, d) is the decommitment of $c2$ to a solution s of the puzzle Y. 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_{ZAP}, or more precisely the witness used to compute the second message b_{ZAP}. In $H_2(v)$, the witness used is $(v, d1)$ whereas in $H_3(v)$ the witness used is (s, d).

First, we show that Invariant 2 holds in $H_3(v)$.

Claim 6. *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 $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists $v \in \{v_0, v_1\}$, a polynomial p and a right interaction k such that k is successful and the value $(\tilde{s}'_k, \tilde{d}'_k)$, extracted from $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$, 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 $H_3(v)$ with A as follows:

- Step 1: On receiving the first message $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a solution s to the puzzle Y by running Sol.

- Step 2: B computes commitments $(c1, c2, c3, b_{\mathsf{NM}})$ (as in $H_3(v)$). Let $d1$ be the decommitment string of the commitment $c1$, which commits to the value v, and d be the decommitment string of the commitment $c2$, which commits to the solution s.

- Step 3: B sends a_{ZAP} as the first message in the WI game of ZAP with the statement $x = (Y, c1, c2, c3, a_{\mathsf{NM}}, b_{\mathsf{NM}})$ and witnesses $w_0 = (v, d1)$ and $w_1 = (s, d)$. 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 $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as the second 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 $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$ and outputs 1 iff the extracted value $(\tilde{s}'_k, \tilde{d}'_k)$ is a fake witness (i.e., B outputs 1 iff $\tilde{s}'_k$ is a solution of the puzzle $\tilde{Y}_k$ and $\mathsf{EOpen}_2(\tilde{c2}_k, \tilde{s}'_k, \tilde{d}'_k) = 1$).

It is easy to see that if the second message b_{ZAP} of ZAP is computed using the witness $w_0 = (v, d1)$ then B perfectly emulates $H_2(v)$ for A and if the second message b_{ZAP} of ZAP is computed using the witness $w_1 = (s, d)$ then B perfectly emulates $H_3(v)$ 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 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 $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$, $o\mathcal{E}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{d'_{\mathsf{NM}},S_2}$, and the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + +\mathsf{size}(\mathsf{Sol}) + \mathsf{size}(o\mathcal{E}_{\mathsf{NM}}) + \mathsf{poly}(n)\\ &\leq \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(S_2)\\ &< \mathsf{poly}(S^*) \qquad \text{(since, } S^* >> S_2, S_1, S_4 \text{ from Equation (7))}\end{aligned}$$

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

Claim 7. *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 $D \in \mathcal{P}/\text{poly}$ 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 $B \in \mathcal{C}_{S^*}$ 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 6.

B with v and k hard-wired in it, participates in the WI game of ZAP and internally emulates an execution of $H_3(v)$ with A as follows:

- Steps 1,2,3 and 4 are identical to the circuit described in Claim 6.
- Step 5: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\tilde{c1}_i$ to extract values $\tilde{v}'_i$. For every unsuccessful right interaction i, B sets $\tilde{v}'_i = \bot$.
- Step 6: B then runs D with the view of A and the values $\{\tilde{v_i}'\}_{i\in[m]}$ 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 $w_0 = (v, d1)$ then B perfectly emulates $H_2(v)$ for A and if the second message b_{ZAP} of ZAP is computed using the witness $w_1 = (s, d)$ then B perfectly emulates $H_3(v)$ for A. Moreover, for every successful interaction i, B sets $\tilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$ and for every unsuccessful interaction, it sets $\tilde{v}'_i = \bot$. 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 $B \in \mathcal{C}_{S^*}$: Apart from running A and Sol, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\tilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(\mathsf{Sol}) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n)\\ &\le \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(n) \cdot \mathsf{poly}(S_{\mathsf{NM}})\\ &< \mathsf{poly}(S^*) \qquad (\text{since, } S^* >> S_2, S_4, S_{\mathsf{NM}}, S_1 \text{ from Equation (7)})\end{aligned}$$

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

Hybrid $H_4(v)$: Hybrid $H_4(v)$ proceeds identically to $H_3(v)$ except that the ECom_3 commitment $c3$ sent to A in the left interaction is generated differently. In $H_3(v)$ $c3$ is committing to the decommitment $(v, d1)$ of $c1$ whereas in $H_4(v)$ $c3$ is committing to the decommitment (v_0, d_0) of $c1^*$. More precisely, $H_4(v)$ computes $(c1, c2, b_{\mathsf{NM}})$ identical to $H_3(v)$. Furthermore, it computes another ECom_1 commitment $c1^*$ which commits to v_0. Let d_0 be the corresponding decommitment string. Then, $H_4(v)$ computes the ECom_3 commitment $c3$ to commit to the decommitment (v_0, d_0) of $c1^*$. The rest of the execution is simulated identically to $H_3(v)$. We note that only difference between hybrids $H_3(v)$ and $H_4(v)$ is the ECom_3 commitment $c3$ which in $H_3(v)$ commits to the decommitment of $c1$ (to the value v) whereas in $H_4(v)$ $c3$ commits to a decommitment of $c1^*$ (to the value v_0).

First, we show that Invariant 2 holds in $H_4(v)$.

Claim 8. *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 $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. Let us assume for contradiction that there exists $v \in \{v_0, v_1\}$, a polynomial p and a right interaction k such that k is successful and the value $(\tilde{s}'_k, \tilde{d}'_k)$, extracted from $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$, 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 $H_4(v)$ with A as follows:

- Step 1: On receiving the first message $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, B obtains a solution s to the puzzle Y by running Sol.

- Step 2: It computes $(c1, c2, b_{\mathsf{NM}})$ as in $H_4(v)$. Let $d1$ be the decommitment string of the commitment $c1$.

- Step 3: It computes a commitment $c1^*$ to the (fixed) value v_0 using ECom_3. Let d_0 be the corresponding decommitment string.

- Step 4: In the hiding game of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$, B sends $(v, d1)$ and (v_0, d_0) as challenges and receives a commitment c^* to either $(v, d1)$ or (v_0, d_0).

- Step 5: B generates the second message of ZAP (b_{ZAP}) by setting $c3 = c^*$. It then sends $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as second round message in the left interaction to A.

- Step 6: 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 $(\tilde{a}_{\mathsf{NM}k}, \tilde{b}_{\mathsf{NM}k})$ and outputs 1 iff the extracted value $(\tilde{s}'_k, \tilde{d}'_k)$ is a fake witness (i.e., B outputs 1 iff $\tilde{s}'_k$ is a solution of the puzzle $\tilde{Y}_k$ and $\mathsf{EOpen}_2(\widetilde{c2}_k, \tilde{s}'_k, \tilde{d}'_k) = 1$).

It is easy to see that if B receives a commitment to $(v, d1)$, then it perfectly emulates $H_3(v)$ for A and if it receives a commitment to (v_0, d_0) then it perfectly emulates $H_4(v)$ for A. By Claim 6, 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 hiding of ECom_3.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_3,S_3}$: B internally runs $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$, $o\mathcal{E}_{\mathsf{NM}} \in \mathcal{C}^{\wedge}_{d'_{\mathsf{NM}},S_2}$, and the rest of the computation done by B takes $\mathsf{poly}(n)$ time. Thus, we have,

$$\begin{aligned}
\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(\mathsf{Sol}) + \mathsf{size}(o\mathcal{E}_{\mathsf{NM}}) + \mathsf{poly}(n) \\
&\leq \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(S_2) \\
&< \mathsf{poly}(S_3) \qquad \text{(since, } S_3 >> S_2, S_1, S_4 \text{ from Equation (7))}
\end{aligned}$$

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. □

Claim 9. *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 $v \in \{v_0, v_1\}$, a distinguisher $D \in \mathcal{P}/\text{poly}$ and a polynomial p 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 $B \in \mathcal{C}^{\vee}_{d_3,S_3}$ 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 8.

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 $H_4(v)$ with A as follows:

- Steps 1-5 are identical to the hiding circuit described in Claim 8.
- Step 6: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_1$ on $\tilde{c1}_i$ to extract values $\tilde{v}'_i$. For every unsuccessful right interaction i, B sets $\tilde{v}'_i = \perp$.
- Step 7: B then runs D with the view of A and the values $\{\tilde{v}_i{}'\}_{i\in[m]}$ 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 $H_3(v)$ for A and if it receives a commitment to $(v_0, d0)$ then it perfectly emulates $H_4(v)$ for A. Moreover, B for every successful interaction i, sets $\tilde{v}'_i$ to the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$ and for every unsuccessful interaction, it sets $\tilde{v}'_i = \perp$. 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 $B \in \mathcal{C}^{\vee}_{d_3,S_3}$: Apart from running A and Sol, B runs $o\mathcal{E}_1$ on at most $m = \mathsf{poly}(n)$ commitments $\tilde{c1}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$ and $o\mathcal{E}_1 \in \mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$, we have,

$$\begin{aligned}\mathsf{size}(B) &= \mathsf{size}(A) + \mathsf{size}(\mathsf{Sol}) + m \cdot \mathsf{size}(o\mathcal{E}_1) + \mathsf{poly}(n)\\ &\le \mathsf{poly}(S_4) + \mathsf{poly}(S_1) + \mathsf{poly}(n)\cdot \mathsf{poly}(S_{\mathsf{NM}})\\ &< \mathsf{poly}(S_3) \qquad \text{(since, } S_3 >> S_4, S_{\mathsf{NM}}, S_1 \text{ from Equation (7))}\end{aligned}$$

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 $H_5(v)$: Hybrid $H_5(v)$ proceeds identically to $H_4(v)$ except that the ECom_1 commitment $c1$ sent to A in the left interaction is generated differently. In $H_4(v)$, $c1$ is committing to the value v whereas in $H_5(v)$ $c1$ is committing to the value (fixed) v_0 instead. 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_1 commitment $c1$ which in $H_4(v)$ commits to v but in $H_5(v)$ $c1$ commits to v_0.

First, we show that Invariant 2 holds in $H_5(v)$.

Claim 10. *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 $(\tilde{a}_{\mathsf{NM}i}, \tilde{b}_{\mathsf{NM}i})$ is a fake witness, is negligible.*

Proof. We claim that $H_5(v)$ is identical to $H_4(v_0)$. Then by Claim 8, Claim 10 holds. □

Claim 11. *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 $v \in \{v_0, v_1\}$, a distinguisher $D \in \mathcal{P}/\mathrm{poly}$ and a polynomial p such that D distinguishes $\mathsf{emim}^A_{H_4}(v)$ from $\mathsf{emim}^A_{H_5}(v)$ with probability $\frac{1}{p(n)}$. Then using A and D, we construct a non-uniform circuit $B \in \mathcal{C}^{\vee}_{d_1,S_1}$ that violates the hiding of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ with non-negligible advantage $\frac{1}{2p(n)}$.

The circuit B with v, v_0 and k hard-wired in it, participates in the hiding game of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ and internally emulates an execution of $H_5(v)$ with A as follows:

- Step 1: On receiving the first message $(Y, a_{\mathsf{ZAP}}, a_{\mathsf{NM}})$ from A, 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 2: B generates the second message of the left interaction identically to $H_5(v)$ except that it embeds c^* as the ECom_1 commitment in the message. That is, B computes $(c2, c3, b_{\mathsf{NM}})$ as in $H_5(v)$ and then computes the second message of ZAP (b_{ZAP}) by setting $c1 = c^*$. It then sends $(c1, c2, c3, b_{\mathsf{NM}}, b_{\mathsf{ZAP}})$ as second round message in the left interaction to A.

- Step 3: After A terminates, for every successful right interaction i, B runs the extractor $o\mathcal{E}_3$ on $\tilde{c3}_i$ to extract values $(\tilde{v}'_i, \tilde{d1}'_i)$. For every unsuccessful right interaction i, B sets $\tilde{v}'_i = \perp$.

- Step 4: B then runs D with the view of A and the values $\{\tilde{v_i}'\}_{i \in [m]}$ 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 $H_4(v)$ for A and if it receives a commitment to v_0 then it perfectly emulates $H_5(v)$ for A. We claim that 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, except with negligible probability. Since D distinguishes the distributions with probability $1/p(n)$, B wins the hiding game with advantage at least $1/2p(n)$.

Next, we argue that $B \in \mathcal{C}^{\vee}_{d_1,S_1}$: Apart from running A and Sol, B runs $o\mathcal{E}_3$ on at most $m = \mathsf{poly}(n)$ commitments $\tilde{c3}_i$, and the rest of the computation takes polynomial time (includes running D). Since, $A \in \mathcal{C}^{\wedge}_{d_2,S_4}$, $\mathsf{Sol} \in \mathcal{C}^{\wedge}_{d_3,S_1}$ and $o\mathcal{E}_3 \in \mathcal{C}^{\wedge}_{d_1,S'_3}$,

$$\begin{aligned}
\mathsf{dep}(B) &= \mathsf{dep}(A) + \mathsf{dep}(\mathsf{Sol}) + m \cdot \mathsf{dep}(o\mathcal{E}_3) + \mathsf{poly}(n) \\
&\le \mathsf{poly}(d_2) + \mathsf{poly}(d_3) + \mathsf{poly}(n) \cdot \mathsf{poly}(d_1) \\
&< \mathsf{poly}(d_1) \qquad \text{(since, } d_1 >> d_3, d_2 \text{ from Equation (7))}
\end{aligned}$$

Furthermore, $\mathsf{size}(B) < \mathsf{poly}(S^*)$. Therefore, B belongs to the circuit class $\mathcal{C}_{d_1}$ (resp., $B \in \mathcal{C}^{\vee}_{d_1,S_1}$) which contradicts the $\mathcal{C}^{\vee}_{d_1,S_1}$-hiding of $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$.

The only thing remaining to show to conclude the proof is that the input to D is identical to $\mathsf{emim}^A_{H_4}(v)$ (resp., $\mathsf{emim}^A_{H_5}(v)$) even though B passes the values extracted by running $o\mathcal{E}_3$ on $\tilde{c3}_i$ as input to D, instead of running $o\mathcal{E}_1$ on $\tilde{c1}_i$. Moreover, it is sufficient to only consider successful interactions.

For every successful right interaction i, B runs $o\mathcal{E}_3$ on $\tilde{c3}_i$ to obtain $(\tilde{v}'_i, \tilde{d1}'_i)$. We claim that the value $\tilde{v}'_i$ is identical to the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$, except with negligible probability. Since i is successful, by Claim 10 we know that with over-whelming probability A does not commit to a fake witness in i. Then by the soundness of ZAP, A must have proved that the commitments $\tilde{c1}_i$ and $\tilde{c3}_i$ are valid and $\tilde{c3}_i$ commits to a decommitment of $\tilde{c1}_i$. Therefore, by the over-extractability of $(\mathsf{ECom}_3, \mathsf{EOpen}_3)$ the value $(\tilde{v}'_i, \tilde{d1}'_i)$ extracted from $\tilde{c3}_i$ is identical to $\mathsf{val}(\tilde{c3}_i)$ with over-whelming probability, where $\mathsf{val}(\tilde{c3}_i)$ is a decommitment of $\tilde{c1}_i$ — $(\tilde{v}_i, \tilde{d1}_i)$. Next, due to the over-extractability of ECom_1, the value extracted by $o\mathcal{E}_1$ from $\tilde{c1}_i$ is identical to $\mathsf{val}(\tilde{c1}_i) = \tilde{v}_i$. Therefore, the value $\tilde{v}_i$

obtained by B is identical to the value that $o\mathcal{E}_1$ extracts from $\tilde{c1}_i$. This is now sufficient to conclude that the input to D is identical to $\mathsf{emim}^A_{H_4}(v)$ (resp., $\mathsf{emim}^A_{H_5}(v)$) 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.

6.3 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 $\langle \tilde{C}, \tilde{R} \rangle$ for identities of length significantly larger than $t(n)$-bits (i.e., $2^{t(n)-1}$-bits). In their work [DDN00], they show that their transformation results in a commitment scheme that can accomodate significantly larger lengths of identities but degrades concurrent non-malleability w.r.t. commitment to stand-alone non-malleability w.r.t. commitment. Furthermore, their reduction also inccurs a polynomial loss.

The commitment schemes considered in this work are non-malleable w.r.t. extraction and we claim that their transformation can be used to amplifying the length of identities at the cost of degrading concurrent non-malleability w.r.t. extraction to standalone non-malleability w.r.t. extraction. 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 description of the protocol from [DDN00] is given below. We refer to the protocol as the "log-n" trick for brevity.

The committer $\tilde{C}$ and receiver $\tilde{R}$ 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, $\tilde{C}$ gets a private input $v \in \{0,1\}^n$ which is the value to be committed.

- *Commit stage:*

  1. To commit to a value $v \in \{0,1\}^n$, $\tilde{C}$ 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 $0 \leq i \leq t'-1$, $\tilde{C}$ and $\tilde{R}$ 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, id_i). Then we denote by $c = \{c_i\}_{i \in [t']}$ the entire transcript of the interaction.

- *Reveal stage:*

  On receiving the decommitment $(v, \{r_i\}_i, \{d_i\})$, $\tilde{R}$ verifies that

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

  $\tilde{R}$ 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 $\langle \widehat{C}, \widehat{R} \rangle$ be a commitment scheme and $\mathcal{C}$ be a class of circuits that is closed under composition with $\mathcal{P}/\text{poly}$.*

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 $2^{t(n)-1}$ bits.*

2. *If $\langle \widehat{C}, \widehat{R} \rangle$ is concurrent $\mathcal{C}$-non-malleable w.r.t. commitment then $\langle \tilde{C}, \tilde{R} \rangle$ 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 $\langle \tilde{C}, \tilde{R} \rangle$ 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 $\langle \tilde{C}, \tilde{R} \rangle$ follows from the statistical binding of $\langle \widehat{C}, \widehat{R} \rangle$. Furthermore, $\langle \tilde{C}, \tilde{R} \rangle$ as defined above accomodates identities of length $t' = 2^{t(n)-1}$-bits.

- Non-malleability w.r.t. commitment: The proof follows from the proof 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 $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ 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, $t' \leq n$, $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ fails with negligible probability. Moreover, $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$ runs $\widehat{o\mathcal{E}}_{\mathsf{NM}}$ on $t' \leq n$ 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. $\widetilde{o\mathcal{E}}_{\mathsf{NM}}$.

- Non-malleability w.r.t. extraction: The proof follows from the proof presented in [DDN00].

□

7 Concurrent Non-malleable Commitment Scheme for Identities of Length n

In this section, we describe the construction of a concurrent non-malleable commitment scheme that can accomodate n-bit identities. The idea is to start with a basic commitment scheme with some non-malleability properties and then apply the log-n trick from Section 6.3 and the strengthening technique described in Section 6.2 repeatedly. More precisely, we will start with the basic commitment scheme ($\mathsf{ENMCom}, \mathsf{ENMOpen}$) from Section 5 which is one-one non-malleable (w.r.t. extraction) for identities of length $t(n) < n$ and then apply the strengthening technique to construct $\langle C^0, R^0 \rangle$, a commitment scheme which is instead concurrent non-malleable albeit accomodating only $t(n)$-bit identities.

Since, $\langle C^0, R^0\rangle$ is concurrent non-malleable (w.r.t. extraction and commitment), we can apply the log-n trick to increase the length of identities to $2^{t(n)-1}$-bits while degrading concurrent non-malleability to stand-alone non-malleability. The concurrent non-malleability can then be restored by applying the strengthening technique. [11] We refer to applying the log-n trick and the strengthening technique successively as an iteration which is represented by the index j. Let $\langle C^{j-1}, R^{j-1}\rangle$ be the commitment scheme obtained at the end of $j-1$ iterations starting with $\langle C^0, R^0\rangle$. In the jth iteration, the log-n trick is applied to $\langle C^{j-1}, R^{j-1}\rangle$ resulting in the commitment scheme $\langle C^{\tilde{j}-1}, R^{\tilde{j}-1}\rangle$ to which the strengthening technique is applied to construct $\langle C^j, R^j\rangle$. Let $\mathsf{complex}^j(n)$ be the computational complexity of $\langle C^j, R^j\rangle$ and $\mathsf{id}^j(n)$ be the maximum length of identities that $\langle C^j, R^j\rangle$ can accomodate. Then, $\mathsf{id}^0(n) = t(n)$ and $\mathsf{complex}^0(n) = \mathsf{poly}(n)$.

Number of iterations to reach n-bit identites: Due to the log-n trick, the identities in successive iterations satisfy,

$$\mathsf{id}^j(n) = 2^{\mathsf{id}^{j-1}(n)-1} \ . \tag{15}$$

Lin and Pass (see Section 5 in [LP09]) showed that starting with $t(n)$-bit identities, one needs to apply the log-n trick $r(n) = O(\log^* n - \log^* t(n))$ times to reach n-bit identities. More precisely, they show that for $r(n)$ as defined above, $\mathsf{id}^{r(n)}(n) \geq \log n + 1$. Then, assuming that $\langle C^{r(n)}, R^{r(n)}\rangle$ is a commitment scheme on $(\log n + 1)$-bit identities, performing another iteration with $\langle C^{r(n)}, R^{r(n)}\rangle$ will give us the commitment scheme $\langle C^{r(n)+1}, R^{r(n)+1}\rangle$ for n-bit identities.

Efficiency of $\langle C^{r(n)+1}, R^{r(n)+1}\rangle$: Similar to [LP09], we want to start with a commitment scheme $\langle C^0, R^0\rangle$ on $O(1)$-bit identities, that is, $t(n) = O(1)$. This implies that we will need super-constant iterations — $O(\log^* n)$. This is always a concern w.r.t. the efficiency of the commitment scheme obtained at the end of super-constant iterations. However, if the complexity function $\mathsf{complex}^j(n)$ in successive iterations were to satisfy the following,

$$\mathsf{complex}^{j+1}(n) = O(\mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n)) + \mathsf{poly}(n) \ , \tag{16}$$

then by the same analysis as in [LP09], we can conclude that $\mathsf{complex}^{r(n)}(n)$ is upper-bounded by some polynomial $\mathsf{poly}(n)$. That is, $\langle C^{r(n)}, R^{r(n)}\rangle$ is efficient and so the commitment scheme $\langle C^{r(n)+1}, R^{r(n)+1}\rangle$ for n-bit identities is also efficient.

For now, let us assume that $\mathsf{complex}^j(n)$ satisfies Equation (16). We will revisit the efficiency of $\langle C^{r(n)}, R^{r(n)}\rangle$ later. Next, we describe the hierarchy of functions d_i and S_k that we will use in our iterations.

Number of functions in the hierarchy: Consider the strengthening technique described in Section 6.2. It upgrades the one-one non-malleability (w.r.t. extraction) of the commitment scheme $\langle C, R\rangle$ to concurrent non-malleability w.r.t. extraction and commitment. The technique relies on the security of other primitives (e.g., $(\mathsf{ECom}_i, \mathsf{EOpen}_i)$) to achieve concurrent non-malleability. The security of these primitives is defined by a hierarchy of non-decreasing functions as shown in Equation (7) (recalled below).

$$\begin{aligned} &d_2 << d_4 << d_3 << d_1 << d_{\mathsf{NM}} << d'_{\mathsf{NM}} << \\ &S_4 << S_1 << S_{\mathsf{NM}} << S_2 << S_3 << S'_3 \ . \end{aligned}$$

[11] Note that the strengthening technique can be applied because the scheme obtained from applying the log-n trick to $\langle C^0, R^0\rangle$ is one-one non-malleable w.r.t. extraction.

More precisely, given that $\langle C, R\rangle$ is $\mathcal{C}^{\wedge}_{d_{\mathsf{NM}},S_{\mathsf{NM}}}$ one-one non-malleable (w.r.t. extraction) and (d'_{NM}, S_2)-over-extractable, the technique considers functions d_1, d_2, d_3, d_4, S_1, S_4, S_3 and S'_3 as above. Then, considering appropriate primitives with their security based on above described functions, the strengthening technique produces the commitment scheme $\langle \widehat{C}, \widehat{R}\rangle$ which is concurrent non-malleable (w.r.t extraction and commitment) but only against circuits in the class $\mathcal{C}^{\wedge}_{d_2,S_4}$. Therefore, the strengthening technique uses four auxiliary functions corresponding to depths (d_1, d_2, d_3 and d_4) and four corresponding to sizes (S_1, S_4, S_3 and S'_3) to upgrade to concurrent non-malleability. Note that while applying the log-n trick, no such auxiliary functions are used. Therefore, in an iteration, we end up using four auxiliary functions each for depths and sizes.

Infact, a careful observation enables us to fix the functions S_2, S_3 and S'_3 through all the iterations. This implies that for the strengthening technique we need four auxiliary functions correpsonding to depths and four auxiliary functions corresponding to size for the first application of the strengthening technique. For successive iterations, we only need two size functions in addition to four depth functions.

As mentioned before, $\langle C^0, R^0\rangle$ only accomodates $O(1)$-bit identities, that is $t(n) = O(1)$. Then, let $r = r(n)+1$ be the number of times we have to apply the transformation to the scheme $\langle C^0, R^0\rangle$ accomodating $O(1)$-bit identities to construct $\langle C^r, R^r\rangle$ that accomodates n-bit identities. Next, we define a $O(r)$ hierarchy of depth and size functions and clearly define the functions used in each iteration j, where r is the number of iterations to be applied to reach n-bit identities.

7.1 Instantiations

Consider the following hierarchy of functions,

$$\begin{aligned} &n << d_1 << d_2 << \ldots << d_{4r} << d_{4r+1} << \ldots << d_{4r+13} \\ &\quad << S_1 << S_2 << \ldots << S_{2r} << S_{2r+1} << \ldots << S_{2r+13} << S_{2r+14} = S^* \ . \end{aligned} \tag{17}$$

such that for each $i \in \{1, \ldots, 4r+12\}$ and $k \in \{1, \ldots, 2r+12\}$,

- 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.
- there exists a size-robust commitment scheme $(\mathsf{ECom}_{S_k}, \mathsf{EOpen}_{S_k})$ that is $\mathcal{C}_{S_k}$-hiding and $(\mathsf{poly}(n), S_{k+1})$-over-extractable.

We next describe the functions in the hierarchy that are used to instantiate the basic commitment schemes $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ and $\langle C^0, R^0\rangle$. Then, we describe the functions used in all iterations $j \in \{1, \ldots, r\}$.

Instantiating $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ for 3-bit identities: The scheme $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ from Section 5 is constructed to accomodate $\log l$-bit identities from a family of $2l+2$ non-decreasing functions $\bar{d}_0, \ldots, \bar{d}_l, \bar{S}_0, \ldots, \bar{S}_l$. We set $l = 8$ and therefore $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ accomodates 3-bit identities. We set the functions $\bar{d}_0, \ldots, \bar{d}_8, \bar{S}_0, \ldots, \bar{S}_8$ as,

$$\begin{aligned} \bar{d}_0 = d_{4r+5},\ \ldots,\ \bar{d}_8 = d_{4r+13}\ , \\ \bar{S}_0 = S_{2r+3},\ \ldots,\ \bar{S}_8 = S_{2r+11}\ . \end{aligned} \tag{18}$$

Therefore, $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ is non-malleable against circuits of depth at most $\mathsf{poly}(\bar{d}_0) = \mathsf{poly}(d_{4r+5})$ and size at most $\mathsf{poly}(\bar{S}_0) = \mathsf{poly}(S_{2r+3})$. Furthermore, it is (d_{4r+13}, S_{2r+11})-over-extractable.

Constructing $\langle C^0, R^0 \rangle$: Given $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ which is one-one non-malleable against circuits of size at most $\mathsf{poly}(d^0_{\mathsf{NM}})$ and size at most $\mathsf{poly}(S^0_{\mathsf{NM}})$, we apply the strengthening technique to construct the commitment scheme $\langle C^0, R^0 \rangle$. Recall that $(\mathsf{ENMCom}, \mathsf{ENMOpen})$ is $(d'^0_{\mathsf{NM}}, S^0_2)$-over-extractable where,

$$d^0_{\mathsf{NM}} = d_{4r+5} \ , \ S^0_{\mathsf{NM}} = S_{2r+3} \ , \ d'^0_{\mathsf{NM}} = d_{4r+13} \ , \ S^0_2 = S_{2r+11} \ . \tag{19}$$

The auxiliary functions used by the strengthening technique are as follows,

$$\begin{aligned} &d^0_2 = d_{4r+1}, \ d^0_4 = d_{4r+2}, \ d^0_3 = d_{4r+3}, \ d^0_1 = d_{4r+4}, \\ &S^0_4 = S_{2r+1}, \ S^0_1 = S_{2r+2}, \ S^0_3 = S_{2r+12}, \ S'^0_3 = S_{2r+13} \ , \end{aligned} \tag{20}$$

where the superscript 0 is to denote the functions used in the strengthening technique applied to $(\mathsf{ENMCom}, \mathsf{ENMOpen})$. More generally, we will use superscript j to denote the functions used in the jth iteration. Therefore, $\langle C^0, R^0 \rangle$ is non-malleable against circuits with depth at most $\mathsf{poly}(d_{4r+1})$ and size at most $\mathsf{poly}(S_{2r+1})$. Furthermore, it is (d_{4r+13}, S_{2r+11})-over-extractable. [12]

Functions used in the jth iteration: Given $\langle C^{j-1}, R^{j-1} \rangle$ which is non-malleable against circuits of depth at most $\mathsf{poly}(d_{4(r-(j-1))+1})$ and size at most $\mathsf{poly}(S_{2(r-(j-1))+1})$ and which is (d_{4r+13}, S_{2r+11})-over-extractable, that is,

$$d^j_{\mathsf{NM}} = d_{4(r-(j-1))+1} \ , \ S^j_{\mathsf{NM}} = S_{2(r-(j-1))+1} \ , \ d'^j_{\mathsf{NM}} = d_{4r+13} \ , \ S^j_2 = S_{2r+11} \ . \tag{21}$$

To construct $\langle C^j, R^j \rangle$ we set the depths and sizes functions used in the strengthening technique as follows,

$$\begin{aligned} &d^j_2 = d_{4(r-j)+1}, \ d^j_4 = d_{4(r-j)+2}, \ d^j_3 = d_{4(r-j)+3}, \ d^j_1 = d_{4(r-j)+4}, \\ &S^j_4 = S_{2(r-j)+1}, \ S^j_1 = S_{2(r-j)+2}, \ S^j_3 = S_{2r+12}, \ S'^j_3 = S_{2r+13} \ . \end{aligned} \tag{22}$$

The resulting scheme $\langle C^j, R^j \rangle$ is concurrent non-malleable against circuits of depth at most $\mathsf{poly}(d^j_2) = \mathsf{poly}(d_{4(r-j)+1})$ and size at most $\mathsf{poly}(S^j_4) = \mathsf{poly}(S_{2(r-j)+1})$ and is (d_{4r+13}, S_{2r+11})-over-extractable.

Finally, it is easy to see that at the end of r iterations, the resulting commitment scheme $\langle C^r, R^r \rangle$ accomodates n-bit identities and is concurrent non-malleable w.r.t. commitment against circuits of depth at most $\mathsf{poly}(d_1)$ and size at most $\mathsf{poly}(S_1)$. Since $S_1 >> d_1 >> n$, we have that $\langle C^r, R^r \rangle$ is non-malleable against all ciruits in $\mathcal{P}/\text{poly}$.

Description of functions d_i and S_k: We derive the functions d_i and S_k. Let us assume that for some $0 < \delta, \varepsilon < 1$, there exists a $(\mathsf{poly}(2^{t^\delta}), 2^{n^\varepsilon})$-TL puzzle where the hardness holds for difficulty parameters $t > c \log n$ for some sufficiently large constant c. Furthermore, we assume the existence of $\mathsf{poly}(2^{n^\delta})$-secure OWPs. Let $i_{\mathsf{max}} = 4r + 13$ and $k_{\mathsf{max}} = 2r + 14$ be the number of depths and size levels required (see Equation 17) for $r = O(\log^* n)$ iterations. Furthermore, let $t_1(n)$ be the smallest function for which the hardness of TL holds and

$$2^{t_1^\delta} >> n \ . \tag{23}$$

An example of such a function is $t_1(n) = (\log n)^{\frac{2}{\delta}}$. Based on the above defined difficulty parameter function t_1, we define a sequence of depths $\{d_i\}$ and size $\{S_k\}$ functions.

[12] Note that $\langle C^0, R^0 \rangle$ is actually (d_{4r+5}, S_{2r+3})-over-extractable which implies that it is also (d_{4r+13}, S_{2r+11})-over-extractable.

Depth Functions d_i: For $i \in \{1, \ldots, i_{\mathsf{max}}\}$, we set the ith depth function in the hierarchy d_i as,

$$d_i = 2^{(t_i)^\delta} \;, \tag{24}$$

where t_i is the difficulty parameter of the puzzles sampled by ECom_{d_i} and is given by,

$$t_i = (t_1)^{\left(\frac{1}{\delta}\right)^{i-1}} \;. \tag{25}$$

Therefore, it is easy to see that depths d_i as defined in Equation (7.1) satisfy the following,

$$d_i >> d_{i-1} \qquad ; \qquad d_1 = 2^{t_1{}^\delta} >> n \;.$$

Therefore for the hierarchy of depth functions defined above, for every i, there exists $(\mathsf{ECom}_{d_i}, \mathsf{EOpen}_{d_i})$ which is $\mathcal{C}_{d_i}$-hiding and (d_{i+1}, d_{i+1})-over-extractable.

Size Functions S_k: We set the first size function S_1 as follows,

$$S_1 = d_{i_{\mathsf{max}}+1} >> d_{i_{\mathsf{max}}} \;. \tag{26}$$

For every $k \in \{1, \ldots, k_{\mathsf{max}}\}$, we require that there exist a size-robust commitment scheme which is $\mathcal{C}_{S_k}$-hiding and $(\mathsf{poly}(n), S_{k+1})$-over-extractable. Using the concrete construction of size-robust commitment schemes described in Section 4.2, we set the security parameter of the OWP as,

$$n_k = (\log S_k)^{\frac{1}{\delta}} \;, \tag{27}$$

to achieve hiding against $\mathsf{poly}(S_k)$-sized circuits. Therefore, the security parameter n_1 corresponding to S_1 satisfies,

$$n_1 = (\log S_1)^{\frac{1}{\delta}} = t_1{}^{\left(\frac{1}{\delta}\right)^{i_{\mathsf{max}}}} \;. \tag{28}$$

Once we have set n_1, we describe all the security parameters n_k's and thereby describe the functions S_k's. We set n_k as follows,

$$n_k = (n_1)^{\left(\frac{1}{\delta}\right)^{k-1}} = t_1{}^{\left(\frac{1}{\delta}\right)^{i_{\mathsf{max}}+k-1}} \;. \tag{29}$$

Then it is easy to see that $\{S_k\}$ satisfies the following,

$$S_{k+1} >> S_k \;; S_{k+1} \geq 2^{n_k} \;. \tag{30}$$

Finally, we require that $n_{k_{\mathsf{max}}}$ is still upper-bounded by some polynomial. We show the existence of a polynomial upper-bounding $n_{k_{\mathsf{max}}}$ below.

$$n_{k_{\mathsf{max}}} = t_1{}^{\left(\frac{1}{\delta}\right)^{i_{\mathsf{max}}+k_{\mathsf{max}}-1}} \;.$$

Since, both $i_{\mathsf{max}}, k_{\mathsf{max}} \in O(\log^* n)$,

$$n_{k_{\mathsf{max}}} = t_1{}^{\left(\frac{1}{\delta}\right)^{O(\log^* n)}} \;.$$

Moreover, $t_1 = (\log n)^{\frac{2}{\delta}}$ and the exponent belongs to $O(\log \log n)$. Therefore,

$$n_{k_{\mathsf{max}}} = (\log n)^{O(\log \log n)} = \mathsf{poly}(n) \;.$$

Therefore, assuming the existence of $(2^{t^\delta}, 2^{n^\varepsilon})$-TL puzzles and $\mathsf{poly}(2^{n^\delta})$-secure OWPs, we have defined a hierarchy of non-decreasing functions as required in Equation (17).

7.2 Efficiency Revisited

We had mentioned above that if the computational complexity $\mathsf{complex}^{j+1}(n)$ of the scheme $\langle C^{j+1}, R^{j+1}\rangle$ obtained from $\langle C^j, R^j\rangle$ satisfies Equation (16) then the complexity of the commitment scheme obtained at the end of super-constant (i.e., $O(\log^* n)$) iterations is upper bounded by a polynomial. However, our transformation (log-n trick + strengthening technique) in its current form does not satisfy Equation (16). More precisely, let $\widetilde{\mathsf{complex}}^j(n)$ denote the complexity of the $\langle \tilde{C^j}, \tilde{R^j}\rangle$ obtained by applying the log-n trick to $\langle C^j, R^j\rangle$. Then it is easy to see that,

$$\widetilde{\mathsf{complex}}^j(n) = \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n)\ . \tag{31}$$

After applying our strengthening technique to $\langle \tilde{C^j}, \tilde{R^j}\rangle$, the resulting complexity $\mathsf{complex}^{j+1}(n)$ is such that,

$$\begin{aligned}\mathsf{complex}^{j+1}(n) &= \mathsf{poly}(n) + \widetilde{\mathsf{complex}}^j(n) + \mathsf{poly}(\widetilde{\mathsf{complex}}^j(n))\\ &= \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n) + \mathsf{poly}(\mathsf{complex}^j(n))\ ,\end{aligned} \tag{32}$$

since $\mathsf{id}^{j+1}(n) \leq n$ and $\mathsf{complex}^j(n) \geq n$. For $\mathsf{complex}^{j+1}(n)$ as defined above and $r = O(\log^* n)$, it is infeasible to find a polynomial $\mathsf{poly}(n)$ such that the complexity of the scheme $\langle C^r, R^r\rangle$ is upper-bounded by $\mathsf{poly}(n)$, that is, $\mathsf{complex}^r(n) < \mathsf{poly}(n)$. The blow up in the complexity from polynomial (i.e., $\mathsf{complex}^0(n) = \mathsf{poly}(n)$) to super-polynomial is due to the term $\mathsf{poly}(\mathsf{complex}^j(n))$ in the expression of $\mathsf{complex}^{j+1}(n)$ which corresponds to the complexity of computing the ZAP proof in our strengthening technique. Recall that the ZAP proof proves the following OR-statement — (a) either $c1$ and $c3$ are valid and $c3$ commits to a decommitment of $c1$ (b) or $c2$ and $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ are valid and $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ commits to a decommitment of $c2$ to a solution of the puzzle Y, where the commitment $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ is generated using $\langle \tilde{C^j}, \tilde{R^j}\rangle$. Since, the reveal phase (checking the validity) of $\langle \tilde{C^j}, \tilde{R^j}\rangle$ runs in time at most $\widetilde{\mathsf{complex}}^j(n)$, the corresponding ZAP statement has length $O(\widetilde{\mathsf{complex}}^j(n))$, thereby contributing the term $\mathsf{poly}(\mathsf{complex}^j(n))$ in the expression for $\mathsf{complex}^{j+1}(n)$. The polynomial dependence of $\mathsf{complex}^{j+1}(n)$ on $\mathsf{complex}^j(n)$ can be reduced by designing a more efficient validity check for $\langle \tilde{C^j}, \tilde{R^j}\rangle$ allowing ZAP to then run only in $\mathsf{poly}(n)$ time.

We note that the commitment schemes generated by our strengthening technique (and then by the log-n trick) are such that the reveal stage of $\langle C^j, R^j\rangle$ can be decomposed into two functions: $\mathsf{PrivOpen}^j$ and $\mathsf{PubOpen}^j$ such that the $\mathsf{PubOpen}^j$ runs independent of the private decommitment information while $\mathsf{PrivOpen}^j$ depends on the decommitment information but is very efficient. Using these two functions, one can then modify the strengthening technique to run the PubOpen function at the end of the commit stage and ask ZAP to prove that PrivOpen accepts the commitment $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$. Then ZAP's complexity depends on the running time of PrivOpen and not on $\mathsf{complex}^j(n)$. Furthermore, we show for all iterations j, there exists a universal polynomial $t^*(n)$ that upper-bounds the running time of PrivOpen. The complexity function $\mathsf{complex}^{j+1}(n)$ then satisfies,

$$\mathsf{complex}^{j+1}(n) = \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n) + \mathsf{poly}(t^*(n))\ ,$$

implying that $\mathsf{complex}^r(n) = \mathsf{poly}(n)$ for some polynomial $\mathsf{poly}(n)$.

Next, we build some notation, describe modifications to the log-n trick and our strengthening technique and show the existence of the universal polynomial $t^*(n)$.

Open-decomposability of a commitment scheme: Let $\langle C, R\rangle$ be a commitment scheme where C and R interact in the *Commit* phase to generate a commitment c. Then in the *reveal* phase C sends private inputs (v, d) as decommitment to R which R accepts iff $\mathsf{Open}(c, v, d) = 1$.

Consider two functions $\mathsf{PubOpen}$ and $\mathsf{PrivOpen}$: $\mathsf{PubOpen}(c)$ is a function which takes as input a commitment c and returns 1/0. $\mathsf{PubOpen}$ executes independently of the private inputs (v, d), hence the name $\mathsf{PubOpen}$. $\mathsf{PrivOpen}(c^*, v, d)$ is a function which takes as input a *small* section c^* of the commitment c along with private inputs (v, d) and returns 1/0.

Definition 18 (t-open-decomposability)**.** *We say that a commitment scheme $\langle C, R\rangle$ is t-open-decomposable w.r.t.* $(\mathsf{PubOpen}, \mathsf{PrivOpen})$ *if the following hold:*

1. $\mathsf{PubOpen}$ *is polynomial time computable.*

2. $\mathsf{PrivOpen}$ *is computable in time t.*

3. *For all $c = c' || c^*$, v and d,*

$$\mathsf{Open}(c, v, d) = 1 \iff \mathsf{PubOpen}(c) = 1 \land \mathsf{PrivOpen}(c^*, v, d) = 1 \ , \tag{33}$$

where Open *is the function executed by R in the reveal phase to accept/reject a commitment.*

Note that any commitment scheme $\langle C, R\rangle$ with open function Open is t-open-decomposable for a sufficiently large t with $\mathsf{PubOpen}$ being the function that always returns 1 and $\mathsf{PrivOpen}$ being the function Open.

Using the open-decomposability notation, we next discuss the modifications to be made to the log-n trick and the strengthening technique. We consider the commitment scheme $\langle C^j, R^j\rangle$ and assume that it is $\mathsf{t}^j(n)$-open-decomposable. We consider applying the log-n trick and strengthening technique to $\langle C^j, R^j\rangle$ and transform it to the commitment scheme $\langle C^{j+1}, R^{j+1}\rangle$.

We modify the log-n trick as below and show that the resulting scheme is also open-decomposable.

Modifications to log-n trick described in Section 6.3: Let $\langle C^j, R^j\rangle$ be concurrent non-malleable (w.r.t. commitment and extraction) for $\mathsf{id}^j(n)$-bit identities. Let it be $\mathsf{t}^j(n)$-open-decomposable w.r.t. $(\mathsf{PubOpen}^j, \mathsf{PrivOpen}^j)$. The log-n trick results in a commitment scheme $\langle \tilde{C^j}, \tilde{R^j}\rangle$ which is one-one non-malleable (w.r.t. commitment and extraction) for identities of length $\mathsf{id}^{j+1}(n) = 2^{\mathsf{id}^j(n)-1}$. We show that $\langle \tilde{C^j}, \tilde{R^j}\rangle$ is $\mathsf{id}^{j+1}(n) \cdot \mathsf{t}^j(n)$-open-decomposable w.r.t. $(\mathsf{Pub\tilde{O}pen}^j, \mathsf{Priv\tilde{O}pen}^j)$ described below.

- Commit stage:

  The commit stage is same as earlier but at the end $\tilde{R^j}$ executes $\mathsf{Pub\tilde{O}pen}^j$ which runs $\mathsf{id}^{j+1}(n)$ independent instances of $\mathsf{PubOpen}^j$ and aborts if one of the instances of $\mathsf{PubOpen}^j$ returns 0.

- Reveal stage: In the reveal stage, $\tilde{R^j}$ executes $\mathsf{Priv\tilde{O}pen}^j$ which runs $\mathsf{id}^{j+1}(n)$ instances of $\mathsf{PrivOpen}^j$ and rejects iff one of the instances of $\mathsf{PrivOpen}^j$ returns 0.

Note that the running time of $\mathsf{Priv\tilde{O}pen}^j$ is at most $\mathsf{id}^{j+1}(n) \cdot \mathsf{t}^j(n)$. Therefore, due to the open-decomposability of $\langle C^j, R^j\rangle$ it follows that $\langle \tilde{C^j}, \tilde{R^j}\rangle$ is $\mathsf{id}^{j+1}(n) \cdot \mathsf{t}^j(n)$-open-decomposable w.r.t. $(\mathsf{Pub\tilde{O}pen}^j, \mathsf{Priv\tilde{O}pen}^j)$.

Next, we modify the strengthening technique to be applied to $\langle \tilde{C^j}, \tilde{R^j}\rangle$ and show that the resulting scheme $\langle C^{j+1}, R^{j+1}\rangle$ is also open-decomposable.

Modifications to the strengthening technique described in Section 6.2: Let $\langle \tilde{C^j}, \tilde{R^j} \rangle$ be the one-one non-malleable commitment scheme as above. The strengthening technique transforms $\langle \tilde{C^j}, \tilde{R^j} \rangle$ to a commitment scheme $\langle C^{j+1}, R^{j+1} \rangle$ which is concurrent non-malleable. Furthermore, we show that $\langle C^{j+1}, R^{j+1} \rangle$ is open-decomposable w.r.t. $(\mathsf{PubOpen}^{j+1}, \mathsf{PrivOpen}^{j+1})$.

- Commit stage - First round is the same as before.

- Commit stage - Second round: Steps 1-3 and 5 are same as before.

  4. Given a_{ZAP}, C^{j+1} 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 $\bar{s}$ such that $c2$ is a commitment to $\bar{s}$ and $\widetilde{\mathsf{PrivOpen}}^j$ accepts $(a_{\mathsf{NM}}, b_{\mathsf{NM}})$ as a commitment to a decommitment of $c2$ and $\mathsf{Ver}(1^n, Y, \bar{s}) = 1$.

     C^{j+1} proves the statement (a) by using the witness $(v, d1)$.

- Commit stage - Function $\mathsf{PubOpen}^{j+1}$:

  R^{j+1} aborts iff the ZAP proof is not accepting or $\widetilde{\mathsf{PubOpen}}^j((a_{\mathsf{NM}}, b_{\mathsf{NM}})) = 0$.

- Reveal stage - Function $\mathsf{PrivOpen}^{j+1}$:

  On receiving $(v, d1)$ from $\widehat{C}$, $\widehat{R}$ accepts the decommitment if $\mathsf{EOpen}_1(c1, v, d1) = 1$. Otherwise it rejects.

Let the $t(n)$ be the upper-bound on the running time of ECom_1 for security parameter n. Then, $\mathsf{PrivOpen}^{j+1}$ is a $t(n)$-time computable function. Hence, $\langle C^{j+1}, R^{j+1} \rangle$ is $t(n)$-open-decomposable w.r.t. $(\mathsf{PubOpen}^{j+1}, \mathsf{PrivOpen}^{j+1})$, that is $\mathsf{t}^{j+1}(n) = t(n)$. Infact it is easy to see that for all iterations $j \in \{1, \ldots, r\}$, $\mathsf{PrivOpen}^j$ verifies the validity of a commitment generated using $(\mathsf{ECom}_1, \mathsf{EOpen}_1)$ with security parameter n. Therefore, $\mathsf{t}^j(n) = t(n)$ for $j \in \{1, \ldots, r\}$.

Claim 12. *For $j \in \{1, \ldots, r\}$, the commitment scheme $\langle C^j, R^j \rangle$ is $t(n)$-open-decomposable w.r.t. $(\mathsf{PubOpen}^j, \mathsf{PrivOpen}^j)$ where $t(n)$ is the upper-bound on the running time of EOpen_1 for security parameter n.*

Furthermore, let $\widetilde{\mathsf{complex}}^j(n)$ be the computational complexity of $\langle \tilde{C^j}, \tilde{R^j} \rangle$ and let $\mathsf{PrivOpen}^j$ be $\mathsf{t}^j(n)$-time computable. Then, the computational complexities $\mathsf{complex}^{j+1}(n)$ and $\mathsf{complex}^j(n)$ satisfies the following,

$$\begin{aligned} \widetilde{\mathsf{complex}}^j(n) &= \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n) \ , \\ \widetilde{\mathsf{complex}}^{j+1}(n) &= \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n) + \mathsf{poly}(\mathsf{id}^{j+1}(n) \cdot \mathsf{t}^j(n)) \ . \end{aligned} \tag{34}$$

We claim that there exists a polynomial $t^*(n)$ which upper-bounds the running time of all $\widetilde{\mathsf{PrivOpen}}^j$ for $j \in \{0, \ldots, r-1\}$ where $\widetilde{\mathsf{PrivOpen}}^j$ is the private open function for the commitment scheme $\langle \tilde{C^j}, \tilde{R^j} \rangle$. This would then imply that the complexity function $\mathsf{complex}^{j+1}(n)$ satisfies,

$$\mathsf{complex}^{j+1}(n) = \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n) + \mathsf{poly}(t^*(n)) \ , \tag{35}$$

which is the desired expression for $\mathsf{complex}^{j+1}(n)$.

Claim 13. *There exists a universal polynomial $t^*(n)$ such that for every $j \in \{0, \ldots, r\}$, $\langle \widetilde{C^j}, \widetilde{R^j} \rangle$ is $t^*(n)$-open-decomposable w.r.t.* $(\mathsf{Pub\tilde{O}pen}^j, \mathsf{Priv\tilde{O}pen}^j)$.

Proof. Let us consider the commitment scheme $\langle C^0, R^0 \rangle$ which has computational complexity $\mathsf{complex}^0(n) = \mathsf{poly}(n)$. Furthermore, let Open^0 be the corresponding open function. As observed earlier, any commitment scheme is t-open-decomposable for a sufficiently large t, we conclude that $\langle C^0, R^0 \rangle$ is $\mathsf{complex}^0(n)$-open-decomposable where the $\mathsf{PubOpen}^0$ function always returns 1 and the $\mathsf{PrivOpen}^0$ function is the function Open^0. Furthermore, we let $t^*(n) = n \cdot \mathsf{max}(\mathsf{complex}^0(n), t(n))$ where $t(n)$ is the upper-bound on the running time of EOpen_1 with security parameter n.

Base Case - $\mathsf{Priv\tilde{O}pen}^0$: The commitment scheme $\langle \widetilde{C^0}, \widetilde{R^0} \rangle$ is the result of applying the modified log-n trick to $\langle C^0, R^0 \rangle$. We note that $\langle \widetilde{C^0}, \widetilde{R^0} \rangle$ is open-decomposable w.r.t. $(\mathsf{Pub\tilde{O}pen}^0, \mathsf{Priv\tilde{O}pen}^0)$ which are described above in the log-n trick. Furthermore, the running time of $\mathsf{Priv\tilde{O}pen}^0$ is $\mathsf{id}^1(n)$ times the running time of $\mathsf{PrivOpen}^0$. Since $\mathsf{id}^1(n) \leq n$ and running time of $\mathsf{PrivOpen}^0$ is at most $\mathsf{complex}^0(n)$, we have that $n \cdot \mathsf{complex}^0(n) \leq t^*(n)$ upper-bounds the running time of $\mathsf{Priv\tilde{O}pen}^0$. The claim is true in the base case.

General Case - $\mathsf{Priv\tilde{O}pen}^j$ for $j \in \{1, \ldots, r\}$: For any j, consider the commitment scheme $\langle C^j, R^j \rangle$. Furthermore, it is $\mathsf{t}^j(n)$-open-decomposable w.r.t. $(\mathsf{PubOpen}^j, \mathsf{PrivOpen}^j)$. From Claim 12 we know that $\mathsf{t}^j(n) = t(n)$.

On applying the modified log-n trick to $\langle C^j, R^j \rangle$ we obtain a commitment scheme $\langle \widetilde{C^j}, \widetilde{R^j} \rangle$ which is open-decomposable w.r.t. $(\mathsf{Pub\tilde{O}pen}^j, \mathsf{Priv\tilde{O}pen}^j)$. Furthermore, the running time of $\mathsf{Priv\tilde{O}pen}^j$ is bounded by $\mathsf{id}^{j+1}(n)$ times the running time of $\mathsf{Priv\tilde{O}pen}^j$. Since $\mathsf{id}^{j+1}(n) \leq n$ and running time of $\mathsf{PrivOpen}^j$ is at most $t(n)$, we have that $n \cdot t(n) \leq t^*(n)$ upper-bounds the running time of $\mathsf{Priv\tilde{O}pen}^j$. □

Therefore by Claim 13 we have that the complexity $\mathsf{complex}^{j+1}(n)$ satisfies the following,

$$\mathsf{complex}^{j+1}(n) = \mathsf{poly}(n) + \mathsf{id}^{j+1}(n) \cdot \mathsf{complex}^j(n) + \mathsf{poly}(t^*(n)) \ , \tag{36}$$

and hence the resulting commitment scheme $\langle C^r, R^r \rangle$ is efficient. Furthermore, $\langle C^r, R^r \rangle$ accomodates identities of length n-bits and is concurrent non-malleable against all circuits in the class $\mathcal{P}/\mathrm{poly}$.

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.

[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.

[COSV16] 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.

[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.

[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.

[Nao03] Moni Naor. On cryptographic assumptions and challenges (invited talk). In Dan Boneh, editor, *CRYPTO 2003*, volume 2729 of *LNCS*, pages 96–109. Springer, Heidelberg, August 2003.

[Pas11] Rafael Pass. Limits of provable security from standard assumptions. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 109–118. ACM Press, June 2011.

[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 subexponential one-way functions. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 638–655. Springer, Heidelberg, May 2010.

[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.