Simulation-Based Concurrent Non-Malleable Commitments and Decommitments

Rafail Ostrovsky[1], Giuseppe Persiano[2], and Ivan Visconti[2]

[1] Department of Computer Science and Department of Mathematics, UCLA, Los Angeles, CA 90095, USA. rafail@cs.ucla.edu
[2] Dipartimento di Informatica ed Applicazioni, Università di Salerno, 84084 Fisciano (SA), ITALY. {giuper,visconti}@dia.unisa.it

Abstract. In this paper we consider commitment schemes that are secure against *concurrent* man-in-the-middle (cMiM) attacks. Under such attacks, two possible notions of security for commitment schemes have been proposed in the literature: concurrent non-malleability with respect to *commitment* and concurrent non-malleability with respect to *decommitment* (i.e., opening).
After the original notion of non-malleability introduced by [Dolev, Dwork and Naor STOC 91] that is based on the independence of the committed messages, a new and stronger simulation-based notion of non-malleability has been proposed with respect to openings or with respect to commitment [1,2,3,4] by requiring that for any man-in-the-middle adversary there is a stand-alone adversary that succeeds with the same probability. When commitment schemes are used as sub-protocols (which is often the case) the simulation-based notion is much more powerful and simplifies the task of proving the security of the larger protocols.
The main result of this paper is a commitment scheme that is simulation-based concurrent non-malleable with respect to both commitment and *decommitment*. This property protects against cMiM attacks mounted during both commitments and decommitments which is a crucial security requirement in several applications, as in some digital auctions, in which players have to perform both commitments and decommitments. Our scheme uses a constant number of rounds of interaction in the plain model and is the first scheme that enjoys all these properties under the simulation-based definitions.

1 Introduction

Commitment schemes are fundamental two-party protocols and have been used in the design of more complex cryptographic protocols since the early 80's (e.g., for coin flipping [5] and for zero-knowledge for NP [6]).

The basic setting in which commitment schemes are defined only requires the hiding and binding properties. However several different scenarios need stronger notions of commitment schemes. In some application scenarios, one wants to be able to guarantee that an adversary $\mathcal{A}$, playing as a receiver in an execution in

which a honest committer commits to message m, is not able to commit to a related value $\tilde{m}$ to a honest receiver in another execution in which $\mathcal{A}$ plays as a committer. It is easy to observe that the hiding property does not guarantee this extra property. This type of adversary is called a *man-in-the-middle* adversary (as the adversary plays in between two honest players). Commitment schemes secure with respect to these attacks are called *non-malleable* commitments.

Two notions of non-malleable commitments have been considered in the literature. A commitment scheme that is *non-malleable with respect to commitment* (in short NMc), first defined by Dolev, Dwork, and Naor [7] guarantees that no polynomial-time man-in-the-middle adversary $\mathcal{A}$ can *commit* to a message $\tilde{m}$ that is related to the message m committed by the honest committer. Instead, a commitment scheme that is *non-malleable with respect to decommitment* (also known as non-malleable with respect to opening), (in short NMd), first defined by Di Crescenzo, Ishai and Ostrovsky [1] guarantees that after the commitment phase, no polynomial-time man-in-the-middle adversary $\mathcal{A}$, observing the decommitment to m of the honest committer, obtains an advantage to *decommit* its commitment to a message $\tilde{m}$ that is related to m.

The need for non-malleable cryptography has been first pointed out in the seminal paper by Dolev, Dwork and Naor [7] who also gave constructions for non-malleable encryption, non-malleable zero-knowledge proofs and non-malleable commitments. The constructions for non-malleable commitments of [7] required $O(\log k)$ rounds, where k is the security parameter. The non-malleability notion of [7] is based on the independence of the committed/decommitted messages played by the man-in-the-middle with respect to the ones played by the sender.

The first non-interactive non-malleable commitment scheme (in the common random string model) was show by Di Crescenzo, Ishai and Ostrovsky [1] (with further efficiency improvement in [2]). They also introduced a new notion of non-malleability by requiring that for any man-in-the-middle adversary there exists a stand-alone simulator with essentially the same success probability. This new simulation-based notion is stronger than the one of Dolev, Dwork and Naor [7] and is much more useful when a commitment scheme is used as sub-protocol since the security of the larger protocol can be proved more easily by using the simulator associated with the commitment scheme.

The first constant-round non-malleable commitment scheme in the plain model (i.e., without setup assumptions such as a common reference string) has been given by Barak [8] under the assumption of the existence of trapdoor permutations and hash functions that are collision resistant against sub-exponential-time adversaries. Pass and Rosen [3] reduced the assumption to the existence of hash functions that are collision resistant against polynomial-time adversaries using simulation-based definition. Pass and Rosen [3] gave two different simulation-based schemes: one that is NMc and one that is NMd.

More recently, Pass and Rosen [4] have considered *concurrent* man-in-the-middle attacks (cMiM attacks) where the man-in-the-middle can be active in any polynomial number of executions as a receiver and as a committer. A commitment scheme that is secure against cMiM attacks is called *concurrent non-*

malleable. As before, we can have two notions of concurrent non-malleable commitment schemes: concurrent NMc and NMd commitment schemes. Pass and Rosen in [4] showed that the NMc scheme of [3] is actually a simulation-based *concurrent* NMc. This implies that simulation-based security is guaranteed if the commitments are concurrently executed but decommitments are not. Their paper leaves as an open problem the construction of constant-round commitment schemes that are simulation-based concurrent NMd. The scheme of [4] enjoys a weaker notion of non-malleability with respect to decommitment that only focuses on the independence of the opened messages [7].

The security of the scheme of [4] relies on the assumption that commitments and decommitments do not overlap in time. We retain this assumption in our schemes. This assumption is motivated by the fact that several important applications have such a separation (e.g., electronic auctions where first all parties send their hidden bids, and only in a second phase they decommit their bids).

Our results. Our main result consists in the construction in the plain model of a constant-round commitment scheme that is simultaneously concurrent NMc and NMd under the simulation-based definition of [3,4]. This implies that security is preserved when polynomially many commitment phases are concurrently executed and when subsequently polynomial many decommitment phases are concurrently executed. This solves a problem left open by the results of [4] and allows one to securely run some commitment-based applications (e.g., digital auctions) by only requiring a constant number of rounds. We follow [4] in that concurrent non-malleability is guaranteed only if commitments and decommitments do not overlap in time (which is the case in several applications).

Our scheme builds and extends multiple techniques. In particular, our scheme uses the perfect NMZK argument of knowledge of [3,4,9] but in a critically different manner. Indeed, whereas in [3,4] the perfect NMZK argument of knowledge is simply combined with a (potentially malleable) commitment scheme and a signature scheme, to achieve security in a concurrent setting we also employ a technique by Feige [10] and a more sophisticated rewind technique. Furthermore, the simulator used by [3,4] works in a straight-line fashion including non-black-box techniques. Our result, instead, combines the straight-line simulation with a new rewinding simulation that still avoids the well known problems of using rewinds in concurrent settings [11]. Our approach also includes and extends some of the techniques developed for building concurrent NMZK in the bare public-key model [12,13]. Finally we stress that in [3] non-malleability with respect to commitment is considered only with respect to statistically binding commitments. Here, perhaps somewhat surprisingly, we show that it is possible to have non-malleable commitments with respect to commitments that are not statistically binding. This is crucially used in our main result since the constant-round NMc and NMd commitment scheme that we show is not statistically binding.

We also remark that the recent work of Barak et al. [14] obtains concurrent non-malleable zero-knowledge with a poly-logarithmic round complexity, and thus does not seem to help for achieving constant-round simulation-based concurrent non-malleable commitments.

2 Simulation-Based Non-Malleable Commitments

Since all our results concern the simulation-based notion of non-malleability, we will concentrate on this notion only. Here we start by considering concurrent non-malleable commitment schemes; that is, commitments schemes that are secure under *Concurrent Man-in-the-Middle* attacks (cMiM attacks). Informally speaking, a non-malleable commitment scheme guarantees that the value committed to (or the value that is decommitted) by a polynomial-time adversary $\mathcal{A}$ is independent of the value simultaneously committed (or the value that is decommitted) to $\mathcal{A}$ by a honest committer. We assume that $\mathcal{A}$ has full power over the scheduling of the messages in the two sessions (the one in which $\mathcal{A}$ is a committer and the one in which $\mathcal{A}$ is a receiver). Following [1,2,3,4], we formalize this notion by comparing two executions: the *man-in-the-middle* execution (the MiM execution) and the *simulated* execution. We denote the security parameter by k and consider the concurrent case where the adversary $\mathcal{A}$ receives and send a polynomial number of commitments.

The Dolev-Dwork-Naor notion of non-malleability. Informally speaking, non-malleability with respect to commitment guarantees that the commitment computed by the MiM adversary corresponds to a message that is independent from the one committed to by the honest committer. In [7], dependency of the values m and $\tilde{m}$ has been formalized through a poly-time computable relation $\mathcal{R}$ for which $\mathcal{R}(m, \tilde{m}) = 1$. Specifically, Dolev, Dwork and Naor[7] defined non-malleability with respect to commitment by requiring that for any man-in-the-middle adversary $\mathcal{A}$ and any polynomial time computable relation $\mathcal{R}$, there exists a poly-time stand-alone adversary S whose success probability in committing to a value $\tilde{m}$ so that $\mathcal{R}(m, \tilde{m}) = 1$ is at least as good as $\mathcal{A}$'s success probability. Non-malleability with respect to decommitment [1] instead considers the ability of $\mathcal{A}$ to decommit to a value $\tilde{m}$ that is related to m. Notice that under the definition of [7], if $\mathcal{A}$ is no more likely to commit to a related value than S and the commitment is statistically binding, then $\mathcal{A}$ is also no more likely to decommit to a related value. This is true regardless of whether $\mathcal{A}$ is given the decommitment information or not. So under this definition, any (statistically binding) commitment that is NMc is also NMd.

A (stronger) simulation-based notion of non-malleable commitments. In this work we adopt the simulation-based definition [1,2,3,4], which requires that the value $\tilde{m}$ committed to by S in the stand-alone execution is computationally indistinguishable from the value committed to by $\mathcal{A}$ in the man-in-the-middle execution. To have a meaningful definition of non-malleability with respect to decommitment, since $\mathcal{A}$ obtains the message m committed by the honest sender before decommitting its commitment, S is assumed to obtain m before decommitting its commitment. It is not clear that with respect to the simulation-based definition, non-malleability with respect to commitment still implies non-malleability with respect to decommitment. The problem here is that (unlike in

the [7] definition), one would like the success probability of S (i.e., the probability that the stand-alone simulator playing with a honest receiver correctly completes the decommitment phase) to be only negligibly far from $\mathcal{A}$'s success probability. Indeed, in the $\mathcal{NMC}$ schemes in the plain model of [3,4], the simulator S generates a bogus commitment that is being fed to $\mathcal{A}$. However, after having committed to some value, S is stuck with the bogus value and it is not clear how to enable S to decommit it to $\mathcal{A}$ as m. (In the common reference string (CRS) model, the situation is easier, as CRS could be arranged so that S can use "equivocal" commitments that can be decommitted to any value, while $\mathcal{A}$ forced to use statistically-binding on CRS commitment [1,2]. Here,we concentrate on the plain model without the CRS, and hence this approach does not work).

From the above discussion we have that the constant-round commitment scheme $\mathcal{NMC}$ of [3] that is proved to be NMc, does not seem to be NMd (according to the simulation-based definition of [3]) or, at least, no evidence of this is provided by the proof of [3]. Specifically, the simulator that computes $c = \mathsf{SBCom}(0^k, s)$ in the commitment phase cannot open c as m since the decommitment phase simply consists in the decommitment phase of SBCom which is *statistically* binding. Therefore under the simulation-based notion of non-malleability, the proof that $\mathcal{NMC}$ is an NMc commitment scheme does not seem to extend to prove that $\mathcal{NMC}$ is also NMd. We stress that in [3,4], only the commitment phase is considered for proving NMc, and since the decommitment phase as discussed above is quite problematic, their security proof implicitly requires that the commitment and decommitment phases do not overlap in time.

NMc does not necessarily require statistical binding. When statistically binding commitments are considered, the commitment phase encodes the unique message to which the commitment can be later decommitted. Indeed, even in case the adversarial committer is unbounded there is no way for him to violate the binding property. Since NMc considers the message committed in the commitment phase, the statistical binding property guarantees that this non-malleability notion is well defined, and indeed in [3] the authors consider the notion of NMc only for statistically binding commitment schemes. Intuitively, NMc seems far more problematic in case the scheme is not statistically binding (but only computationally binding), since the commitment phase does not uniquely specify the message that is going to be decommitted. Therefore, the meaning of NMc for an unbounded adversarial committer is unclear. We observe though that NMc commitments are meant to be secure against polynomial-time MiM adversaries for which the computational binding property still holds. It is therefore potentially possible to have a commitment scheme that is not statistically binding (i.e., binding does necessarily hold in case the adversarial committer is unbounded) but however still is NMc as at the end of the commitment phase it is always possible to determine the message committed by the polynomial-time MiM and by the honest sender. Indeed, what we show in this paper, is the commitment schemes that are *not* statistically binding but that are NMc commitment schemes and, at the same time, NMd. To define NMc we will use the concept of "message committed to by an adversary $\mathcal{A}$ during the commitment phase." By this we

mean the following. We will consider commitment schemes in which, for all adversaries $\mathcal{A}$, and for each possible transcript `trans` of the interaction between adversary $\mathcal{A}$ and a honest receiver R such that R accepts the commitment, there exists (statistically) only one message m that is consistent with `trans`; that is, for which there exist random coin tosses that give `trans`. We stress that statistically hiding commitment schemes do not have the above property and thus our definition is not suitable for these commitment schemes.

For lack of space, in the full version of this paper [15] we review the two schemes of [3] for non-malleable commitments: $\mathcal{NMC}c$ that is NMc and $\mathcal{NMC}d$ that is NMd and we also show a commitment scheme that combines $\mathcal{NMC}c$ and $\mathcal{NMC}d$.

3 Simulation-Based cNM Commitments

Following [3,4], we now formalize the concept of a (simulation-based) *concurrent* non-malleable commitment scheme by comparing two executions: the *concurrent man-in-the-middle* execution (the cMiM execution) and the *simulated* execution. We denote the security parameter by k.

The cMiM execution. In the cMiM execution, the cMiM adversary $\mathcal{A}$ is simultaneously participating in $\mathsf{poly}(k)$ left and $\mathsf{poly}(k)$ right interactions.

Consider a cMiM execution in which the cMiM adversary $\mathcal{A}$ with auxiliary information z interacts in the i-th left interaction with a honest committer running on input a message m_i of length $\mathsf{poly}(k)$ and in the right interactions with honest receivers. We denote by $\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(M, z)$, where $M = (m_1, \ldots, m_{\mathsf{poly}(k)})$, the random variable that associates to the cMiM execution a vector $\tilde{M}$ whose i-th component $\tilde{m}_i$ is defined as follows. If the commitment phase of the i-th right interaction ends successfully and its transcript is different from the commitment phase of all the left interactions, then $\tilde{m}_i$ is the message that $\mathcal{A}$ has *committed to* in the i-th right interaction. Otherwise, $\tilde{m}_i = \perp$.

Similarly, we denote by $\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Dec}}(M, z)$ the vector $\tilde{M}$ whose i-th component $\tilde{m}_i$ is the message that $\mathcal{A}$ has *decommitted* in the right interaction. If the i-th right interaction is not successful or its transcript (including commitment and decommitment phase) is identical to the transcript of one of the left interactions then $\tilde{m}_i = \perp$.

The simulated execution. In the simulated execution we have one party S (called the *simulator*) that interacts with $\mathsf{poly}(k)$ honest receivers. S works in two phases: in the commitment phase S receives security parameter 1^k and auxiliary information z and interacts with the honest receivers. We denote by $\mathsf{csis}^{S}_{\mathsf{Com}}(1^k, z)$ the vector $\tilde{M}$ whose i-th component $\tilde{m}_i$ is the value committed to by S if the i-th commitment phase has been successfully completed. Otherwise $\tilde{m}_i$ is set equal to $\perp$.

Once the commitment phases have been completed, S receives input vector M and interacts with the honest receiver to complete the decommitment phase.

We denote by $\mathsf{csis}^S_{\mathsf{Dec}}(M,z)$ the vector $\tilde{M}$ whose i-th component $\tilde{m}_i$ is the value decommitted by S in the i-th decommitment phase if it has been successfully completed. Otherwise $\tilde{m}_i$ is set equal to $\perp$.

We have the following definitions (see also [3,4]).

Definition 1. *A commitment scheme is simulation-based* concurrent non-malleable with respect to commitment *(*a concurrent NMc commitment scheme*) if, for every probabilistic polynomial-time cMiM adversary $\mathcal{A}$, there exists a probabilistic polynomial time simulator S such that following ensembles are computationally indistinguishable:*

$$\{\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(M,z)\}_{M\in(\{0,1\}^{\mathsf{poly}(k)})^{\mathsf{poly}(k)},z\in\{0,1\}^\star} \text{ and } \{\mathsf{csis}^S_{\mathsf{Com}}(1^k,z)\}_{z\in\{0,1\}^\star}.$$

Definition 2. *A commitment scheme is simulation-based* concurrent non-malleable with respect to decommitment *(*a concurrent NMd commitment scheme*) if, for every probabilistic polynomial-time cMiM adversary $\mathcal{A}$, there exists a probabilistic polynomial time simulator S such that the following ensembles are computationally indistinguishable:*

$$\{\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Dec}}(M,z)\}_{M\in(\{0,1\}^{\mathsf{poly}(k)})^{\mathsf{poly}(k)},z\in\{0,1\}^\star}$$

and

$$\{\mathsf{csis}^S_{\mathsf{Dec}}(M,z)\}_{M\in(\{0,1\}^{\mathsf{poly}(k)})^{\mathsf{poly}(k)},z\in\{0,1\}^\star}.$$

3.1 Commitment Scheme $c\mathcal{NMC}d$

In this section we present a constant-round commitment scheme $c\mathcal{NMC}d$ that enjoys both simulation-based concurrent NMc and simulation-based concurrent NMd. We will use a constant-round tag-based perfect NMZK argument of knowledge $\mathsf{nmZK} = \{\mathcal{P}_t, \mathcal{V}_t\}_t$ for all $\mathtt{NP}$ [3], a constant-round witness indistinguishable $(\mathsf{wi}\mathcal{P}, \mathsf{wi}\mathcal{V})$ proof of knowledge (WIPoK) for all $\mathtt{NP}$ [16,17], a non-interactive statistically binding commitment scheme Com and a secure signature scheme $SS = (\mathtt{SG}, \mathtt{Sig}, \mathtt{SVer})$. The most sophisticated tool that we use is obtained from a sequence of works by Pass and Rosen.

Theorem 1 ([3,4,9]). Assume that there exists a family of claw-free permutations. Then for any $\mathtt{NP}$ language L there exists a constant-round tag-based one-left many-right perfect cNMZK arguments of knowledge $\mathsf{nmZK} = \{\langle \mathcal{P}_{\mathtt{tag}}, \mathcal{V}_{\mathtt{tag}}\rangle\}_{\mathtt{tag}}$ for all $\mathtt{NP}$.

According to the above definition, this theorem says that for any efficient one-left many-right concurrent man-in-the-middle adversary $\mathcal{A}$ that is restricted to one left session there exists an efficient simulator S that guarantees: 1) the view (including the left proof and all the right proofs) given in output by S is perfectly indistinguishable from the interaction of $\mathcal{A}$ with honest prover and honest verifiers; 2) the extraction succeeds for all accepting right proofs in which the one-left many-right concurrent man-in-the-middle adversary has used a tag not appearing in the left proof.

See the full version [15] of this paper for details about the other tools.

A description of commitment scheme $c\mathcal{NMC}d$ is found in Figure 1.

Security Parameter: 1^k.
Input to Committer: $m \in \{0,1\}^k$.
Commitment Phase:

$C \to R$: pick $s \in \{0,1\}^k$, set $c = \mathsf{Com}(m,s)$ and send c to R.
$C \to R$: set $(\mathtt{PK}, \mathtt{SK}) \leftarrow \mathtt{SG}(1^k)$ and send $\mathtt{PK}$ to R.
$C \leftrightarrow R$: C executes the code of $\mathcal{P}_{\mathtt{PK}}$ on input c to prove knowledge of $m, s \in \{0,1\}^k$ such that $c = \mathsf{Com}(m,s)$. R executes the code of $\mathcal{V}_{\mathtt{PK}}$ on input c. If $\mathcal{V}_{\mathtt{PK}}$ rejects then R aborts.
$R \to C$: pick $m_0, s_0, m_1, s_1 \in \{0,1\}^k$, set $c_0 = \mathsf{Com}(m_0, s_0), c_1 = \mathsf{Com}(m_1, s_1)$ and send c_0 and c_1 to C.
$R \leftrightarrow C$: R select a random bit b and executes the code of $\mathsf{wi}\mathcal{P}$ on input (c_0, c_1) to prove knowledge of $\hat{m}, \hat{s} \in \{0,1\}^k$ such that $c_0 = \mathsf{Com}(\hat{m}, \hat{s})$ or $c_1 = \mathsf{Com}(\hat{m}, \hat{s})$ using (m_b, s_b) as witness. C executes the code of $\mathsf{wi}\mathcal{V}$ on input (c_0, c_1). If $\mathsf{wi}\mathcal{V}$ rejects then C aborts.
$C \to R$: let $\mathtt{trans}_0$ be the transcript so far with an extra bit 0 at the end. Set $\sigma_0 \leftarrow \mathtt{Sig}(\mathtt{trans}_0, \mathtt{SK})$ and send σ_0 to R.
R: if $\mathtt{SVer}(\mathtt{trans}_0, \sigma_0, \mathtt{PK}) \neq 1$ abort.

Decommitment Phase:

$C \to R$: send m.
$C \leftrightarrow R$: C executes the code of $\mathcal{P}_{\mathtt{PK}}$ on input (c, c_0, c_1) to prove knowledge of $\hat{m}, \hat{s} \in \{0,1\}^k$ such that $c = \mathsf{Com}(m, \hat{s})$ or $c_0 = \mathsf{Com}(\hat{m}, \hat{s})$ or $c_1 = \mathsf{Com}(\hat{m}, \hat{s})$, using (m, s) as witness. R executes the code of $\mathcal{V}_{\mathtt{PK}}$ on input (c, c_0, c_1). If $\mathcal{V}_{\mathtt{PK}}$ rejects then R aborts.
$C \to R$: let $\mathtt{trans}_1$ be the transcript so far with an extra bit 1 at the end. Set $\sigma_1 \leftarrow \mathtt{Sig}(\mathtt{trans}_1, \mathtt{SK})$ and send σ_1 to R.
R: if $\mathtt{SVer}(\mathtt{trans}_1, \sigma_1, \mathtt{PK}) \neq 1$ abort.

Fig. 1. Our concurrent NMc and concurrent NMd commitment scheme $c\mathcal{NMC}d$.

How we achieve concurrent NMd. First of all, we notice that a straight-forward combination of the two commitment schemes of [3] produces a commitment scheme that we call $\mathcal{NMC}d$ that achieves non-malleability with respect to both commitment and decommitment, when concurrency is not considered. In proving the NMd property one crucially relies on the existence of a simulator extractor for the NMZK argument nmZK. If one tries to argue that $\mathcal{NMC}d$ is a concurrent NMd commitment scheme along the same lines, one would need a simulator that simulates concurrent executions; in other words, one would need a concurrent NMZK argument of knowledge. Unfortunately, the existence of a constant-round concurrent NMZK argument system in the plain model is still an open problem.

We use instead a more sophisticated protocol and prove its properties by blending the straight-line simulator of the concurrent NMc commitment scheme of [4] with a sophisticated rewind technique. In using rewinding we have to

be careful as the nested sessions can potentially make the running time super polynomial[3]. Instead, we perform rewinds "in advance," to extract information from the adversary. The simulator is then able to simulate in a straight-line fashion the decommitment phase by using the information extracted by means of rewinds. Our security proof also employs the two-witness technique by [10] and the well known FLS-technique [18].

In somewhat more details, we extend the commitment phase of the concurrent non-malleable commitment scheme of [4] by requiring that the receiver gives a proof of knowledge of a secret. The decommitment phase consists in sending a message and in proving with a NMZK proof that either the message corresponds to the committed one or the sender knows the secret (this is the FLS-technique [18]). Our simulator will extract the secrets of all receivers in the commitment phase and will use them as fake witnesses in the decommitment phase. Note that one could think that the rewinding technique used by the simulator during the commitment phase could blow up its running time since the adversarial receiver could adaptively play different messages when the transcript changes. Fortunately we adopt a non-dangerous rewind technique that does not harm the running time of the simulator. Indeed, the simulator will first play the commitment phase running the honest sender algorithm. Then it will extract the secrets encoded by the receiver in that specific transcript by running an extractor sequentially for each commitment, one-by-one, starting each time from the same transcript. During this extraction procedure the simulator will not be interested in re-committing again or in simulating concurrent sessions, it will simply play again the honest sender procedure in all sessions with the only exception of the one in which it extracts the secret of the receiver. The extracted secret will only be kept in memory by the simulator and will not be used in the commitment phase. Instead, the decommitment phase will be crucially based on the knowledge of the secrets of the receiver, and will allow the simulator to play in straight-line, opening the committed messages as any messages.

We show that an adversary will not be able to use such a secret, since we prove that any successful adversary can be used to break a standard complexity-theoretic assumption by using the two-witness technique of [10] and the non-malleability of nmZK.

In the next section we prove the properties of commitment scheme $c\mathcal{NM}cd$. We will often use the simulation-extractability property of nmZK. Notice that this property is guaranteed only in case the tag used by the adversary is different from the one used by the other parties. Since in our scheme we use as tag the public key of a signature scheme, and since each phase is only correctly completed if there is a signature under that public key of the transcript of the phase, we assume that the simulation-extractability property always holds, since otherwise the security of the signature scheme is broken. We will detail this argument only when we prove the NMc property for the one-left many-right case (see the discussion below the description of Expt_2), in the other cases the argument is quite similar and is omitted.

[3] The study of this problem started with the notion of concurrent zero knowledge [11].

Binding. In the proof of concurrent NMd we show that any man-in-the-middle adversary that completes the commitment phase, can later open that commitment only in one way. This property is even stronger than binding (since the classical adversary for the binding property can not play as receiver) thus that proof properly contains the proof of the binding property.

Hiding. Assume by contradiction that there exists an adversarial receiver $\mathcal{A}$ that, after the commitment phase distinguishes a commitment to m_0 from a commitment to m_1 with non-negligible advantage. We show how to reduce $\mathcal{A}$ to an adversary $\mathcal{A}'$ that breaks the hiding property of Com. Indeed, $\mathcal{A}'$ on input a challenge com (i.e., a commitment of either m_0 or m_1), plays the honest committer algorithm with the following two exceptions: com is sent in the commitment phase and the simulator for $\mathsf{nmZK}_{\mathsf{PK}}$ is used instead of the honest prover algorithm. Since the simulation for $\mathsf{nmZK}_{\mathsf{PK}}$ is perfect, the only chance $\mathcal{A}$ has to guess concerns the value of com. Therefore, $\mathcal{A}'$ by simply giving in output the same bit given in output by $\mathcal{A}$ succeeds in guessing with non-negligible advantage the message committed in com.

Simulation-Based Concurrent NMc. We start by considering the simpler case in which the adversary $\mathcal{A}$ is active in one left commitment and in polynomially many right commitments (a one-left many-right adversary).

The one-left many-right case. For every one-left many-right MiM adversary $\mathcal{A}$, we consider simulator $S(z)$ that internally runs $\mathcal{A}(z)$ and provides $\mathcal{A}$ with a left commitment by executing the code of the honest committer to commit to 0^k (k is the security parameter). For the right commitments instead S relays messages between the polynomially many honest receivers and $\mathcal{A}$. We stress that for NMc we only have to consider the commitment phase.

We now prove that for all messages $m \in \{0,1\}^k$ and all z, we have that

$$\left|\text{Prob}[\ D(m, \mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(m,z)) = 1\ ] - \text{Prob}[\ D(m, \mathsf{csis}^{S}_{\mathsf{Com}}(1^k,z)) = 1\ ]\right|$$

is negligible in k for all distinguishers D. We consider hybrid experiments starting with $\mathsf{Expt}_0(v,z)$.

$\mathsf{Expt}_0(v,z)$ is the experiment in which $\mathcal{A}(z)$ interacts in the left commitment with a honest committer committing to v and with honest receivers in the right commitments. We denote by $\tilde{M}$ the vector whose i-th component $\tilde{m}_i$ is defined as follows. If the i-th right commitment is successfully completed by $\mathcal{A}$ and its transcript differs from the one of the left commitment then $\tilde{m}_i$ is the message $\mathcal{A}$ has committed to[4] in the i-th right commitment. Otherwise $\tilde{m}_i = \perp$. $\mathsf{Expt}_0(v,z)$ returns $D(v,\tilde{M})$. We set $p_0(v,z) = \text{Prob}[\ \mathsf{Expt}_0(v,z) = 1\ ]$. Obviously, we have that for all z, k and $m \in \{0,1\}^k$, $p_0(m,z) = \text{Prob}[\ D(m, \mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(m,z)) = 1\ ]$ and that $p_0(0^k,z) = \text{Prob}[\ D(m, \mathsf{csis}^{S}_{\mathsf{Com}}(1^k,z)) = 1\ ]$.

[4] This is the message that is consistent with the transcript. Since we use a statistically binding commitment scheme there is a unique such message.

To define the next experiment, we observe that $\mathcal{A}$ naturally defines a one-left many-right MiM adversary $\mathcal{A}'$ for nmZK. Specifically, consider the following adversary $\mathcal{A}'$. $\mathcal{A}'(z)$ internally runs $\mathcal{A}(z)$. $\mathcal{A}'$ forwards externally all $\mathcal{A}$'s messages of all the executions of nmZK. For the execution of $(\mathsf{wi}\mathcal{P}, \mathsf{wi}\mathcal{V})$ of each right commitment (here $\mathcal{A}$ acts as a verifier), $\mathcal{A}'$ computes the commitment of two random messages and executes the code of $\mathsf{wi}\mathcal{P}$. For the executions of $(\mathsf{wi}\mathcal{P}, \mathsf{wi}\mathcal{V})$ of the left commitments, $\mathcal{A}'$ executes the code of $\mathsf{wi}\mathcal{V}$. Now let $\mathcal{S}'$ be the simulator-extractor of nmZK for adversary $\mathcal{A}'$.

Experiment $\mathsf{Expt}_1(v, z)$ differs from $\mathsf{Expt}_0(v, z)$ in that we have the simulator $\mathcal{S}'$ for adversary $\mathcal{A}'$ instead of $\mathcal{A}$ that is playing with the honest prover and honest verifiers for nmZK. More precisely, in the left commitment of $\mathsf{Expt}_1(v, z)$, we first compute $\mathsf{com} = \mathsf{Com}(v, s)$ and $(\mathtt{PK}, \mathtt{SK}) = \mathtt{SG}(1^k)$ and then run $\mathcal{S}'$ on input com, tag $\mathtt{PK}$ and z. All other steps (executions of $(\mathsf{wi}\mathcal{P}, \mathsf{wi}\mathcal{V})$ and signatures) are performed just like in $\mathsf{Expt}_0(v, z)$. Let View be the view output by $\mathcal{S}'$ and define vector $\tilde{M}$ as follows. If the i-th right commitment in View is successfully completed and its transcript differs from the one of the left commitment, then set $\tilde{m}_i$ equal to the message committed to (again, this message is unique since Com is statistically binding) by $\mathcal{A}$. Otherwise, set $\tilde{m}_i = \perp$. Finally, $\mathsf{Expt}_1(v, z)$ outputs $D(v, \tilde{M})$. We set $p_1(v, z) = \text{Prob}[\ \mathsf{Expt}_1(v, z) = 1\ ]$. By the perfect NMZK property of nmZK, we have that $p_0(v, z) = p_1(v, z)$ for all v and z.

Experiment $\mathsf{Expt}_2(v, z)$ differs from $\mathsf{Expt}_1(v, z)$ in the way in which vector $\tilde{M}$ (and consequently the output) is computed. Specifically, in $\mathsf{Expt}_2(v, z)$ we set $\tilde{m}_i$ as the message that has been extracted by $\mathcal{S}'$ as part of the witness for the i-th right execution of nmZK. If no message is extracted then $\tilde{m}_i = \perp$. We set $p_2(v, z) = \text{Prob}[\ \mathsf{Expt}_2(v, z) = 1\ ]$.

Denote by $\tilde{\mathtt{PK}}_i$ the signature public key used as a tag for the i-th right execution of nmZK in View and by $\mathtt{PK}$ the signature public key used as a tag for the left execution of nmZK in View. First of all observe that, for all i, if the transcript of the i-th right commitment of View differs from the one of the left commitment then, by the security of the signature scheme, the probability that $\tilde{\mathtt{PK}}_i = \mathtt{PK}$ is negligible. Therefore, for each i, only two cases have non-negligible probability. In the first case the transcript of the i-th right commitment is equal to the one of the left commitment (and thus $\tilde{\mathtt{PK}}_i = \mathtt{PK}$). Then we observe that in this case $\tilde{m}_i = \perp$ both in $\mathsf{Expt}_1(v, z)$ and in $\mathsf{Expt}_2(v, z)$. If instead the transcript of the i-th right commitment differs from the one of the left commitment and $\tilde{\mathtt{PK}}_i \neq \mathtt{PK}$ then, by the extraction properties of $\mathcal{S}'$, the value $\tilde{m}_i$ extracted by $\mathcal{S}'$ is not the value committed to by $\mathcal{A}$ in View with negligible probability. Therefore we conclude that $|p_2(v, z) - p_1(v, z)|$ is negligible for all v and z.

We now conclude the proof by showing that for all k and for all $v \in \{0,1\}^k$, $|p_2(v, z) - p_2(0^k, z)|$ is negligible. Suppose that it is not and thus for infinitely many k there exists $v_k \in \{0,1\}^k$ and z such that $|p_2(v_k, z) - p_2(0^k, z)| \geq 1/\mathsf{poly}(k)$. Then, we can construct the following adversary B that breaks the hiding of Com. B receives $\hat{c}$ that is a commitment to either 0^k or v_k and executes $\mathsf{Expt}_2(v_k, z)$ by setting in the left commitment phase $c = \hat{c}$. We notice that Expt_2 can be executed in polynomial time even though the message committed

to by c in the left interaction is not known. From the output of the experiment B has a non-negligible advantage in guessing the committed bit.

We have shown that both $|p_0(v_k, z) - p_2(v^k, z)|$ and $|p_2(v_k, z) - p_2(0^k, z)|$ are negligible, Using again the same arguments, it follows that $|p_2(0^k, z) - p_0(0^k, z)|$ is negligible. Therefore, we have that $\Big|\text{Prob}[\ D(m, \mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(m, z)) = 1\ ] - \text{Prob}[\ D(m, \mathsf{csis}^{S}_{\mathsf{Com}}(1^k, z)) = 1\ ]\Big| = |p_0(v^k, z) - p_0(0^k, z)|$ is negligible.

The many-left many-right case for concurrent NMc. We now consider the many-left many-right case. For concurrent MiM adversary $\mathcal{A}$, we consider simulator $S(z)$ that runs $\mathcal{A}(z)$ internally and executes the code of the honest committer on input 0^k for all left commitments. For the right interactions, S relays messages between the external receivers and $\mathcal{A}$. Notice that if we have only one left commitment S coincides with the simulator we used for proving non-malleability with respect to one-left many-right MiM.

Assume by contradiction that there exists a distinguisher D that distinguishes $\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(M, z)$ and $\mathsf{csis}^{S}_{\mathsf{Com}}(1^k, z)$. Let $l = \mathsf{poly}(k)$ be the number of left commitments and, for $i = 0, \ldots, l$, consider hybrid experiment $\mathsf{Expt}^{\mathcal{A}}_i$ defined as follows. Let $M = (m_1, \ldots, m_l)$ be a vector of messages. In $\mathsf{Expt}^{\mathcal{A}}_i(M, z)$, adversary $\mathcal{A}$ is run on input z and the j-th honest left committer commits to m_j if $j \leq i$ and to 0^k otherwise. $\mathsf{Expt}^{\mathcal{A}}_i(M, z)$ outputs a vector whose i-th component consists of the messages committed to by $\mathcal{A}$ in the i-th right commitment if it has been successfully completed by $\mathcal{A}$ and if its transcript differs from the transcripts of all the left commitments. If this is not the case then the i-th component of the output of $\mathsf{Expt}^{\mathcal{A}}_i(M, z)$ is set equal to $\perp$. Obviously, for all M and z, $\mathsf{Expt}^{\mathcal{A}}_0(M, z)$ coincides with $\mathsf{csis}^{S}_{\mathsf{Com}}(1^k, z)$ and $\mathsf{Expt}^{\mathcal{A}}_l(M, z)$ with $\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(M, z)$. If there exists a probabilistic polynomial time distinguisher D that distinguishes between $\mathsf{csis}^{S}_{\mathsf{Com}}(1^k, z)$ and $\mathsf{cmim}^{\mathcal{A}}_{\mathsf{Com}}(M, z)$ then there must be $i \in \{0, \ldots l-1\}$ such that D distinguishes the output of $\mathsf{Expt}^{\mathcal{A}}_i(M, z)$ and the output of $\mathsf{Expt}^{\mathcal{A}}_{i+1}(M, z)$. We stress that the only difference between experiment $\mathsf{Expt}^{\mathcal{A}}_i$ and experiment $\mathsf{Expt}^{\mathcal{A}}_{i+1}$ is that in the $(i+1)$-st left commitment of $\mathsf{Expt}^{\mathcal{A}}_i$ we are committing to 0^k (just like the simulator) whereas in $\mathsf{Expt}^{\mathcal{A}}_{i+1}$ we are committing to m_{i+1}. We can therefore construct a successful MiM adversary $\mathcal{A}'$ for the one-left many-right case. Adversary $\mathcal{A}'$ internally runs all left sessions with the only exception of the $(i+1)$-st session that is played either with a honest committer committing to m_{i+1} or with the simulator of the one-left many-right case. Therefore $\mathcal{A}'$ breaks the one-left many-right non-malleability which is a contradiction.

Simulation-Based Concurrent NMd. For every cMiM adversary $\mathcal{A}$, we describe a simulator S that interacts with polynomially many honest receivers and performs with each of them a commitment and a decommitment phase. To satisfy Definition 2, we will show that, for every vector M of messages, S decommits its commitments to a vector $\tilde{M}$ of messages that is indistinguishable from the messages decommitted by $\mathcal{A}$ when interacting on the left with honest committers committing to M.

The simulator. Since now we also have to care about decommitments, we extend the simulator in the following way. S first runs the left and the right commitment phases with $\mathcal{A}$ executing the code of the honest receiver in the right commitment phases and the code of the honest committer on input message 0^k in the left commitment phase. Notice that $\mathcal{A}$ is interacting solely with S and no honest receiver is involved. Then S runs the extractors for all the proofs (both in left and right commitment phases) provided by $\mathcal{A}$ in order to get the corresponding witnesses. More precisely, for each right commitment phase, S runs the extractor of nmZK and we denote by (m_i, s_i) the witness extracted in the i-th right commitment phase; for each left commitment phase, S runs the extractor of the WIPoK and we denote by $(m_{b_i,i}, s_{b_i,i})$, with $b_i \in \{0,1\}$, the witness extracted in the i-th left commitment phase. Extractions are executed sequentially and thus the running time of S is polynomial.

Next, S plays the commitment phases with the honest receivers. S does so by executing the code of the honest committer and using, for the i-th commitment phase, message m_i as input.

After the commitment phases have been completed, S receives vector $M^\star = (m_1^\star, \ldots, m_l^\star)$ and has to perform the decommitment phases with $\mathcal{A}$. S does so by resuming the interactions with $\mathcal{A}$ in the following way. In the left decommitment phase corresponding to the i-th left commitment phase, S uses knowledge of $m_{b_i,i}$ to open the commitment (that was originally computed by S as a commitment to 0^k) to $m_i^\star$. In the right decommitment phases, S acts as a honest receiver. Then, for each i, if $\mathcal{A}$ has successfully completed the i-th right decommitment phase, then S completes the i-th decommitment phase with the honest receiver decommitting the commitment to m_i (notice that in the i-th commitment phase with honest receivers, S had committed to m_i). This ends the description of the simulator S.

The above simulator combines the techniques we propose in this paper to overcome the limitations of the [4] result. Our simulator not only guarantees concurrent NMc as we proved previously, but it will also guarantee concurrent NMd. Notice that the [4] simulator only works for concurrent NMc, while for NMd it immediately fails when a single decommitment phase is executed. We now turn to proving that the described simulator S satisfies Definition 2.

We now prove that the distribution of the messages decommitted by $\mathcal{A}$ when interacting with honest committers and honest receivers is indistinguishable from the distribution of the messages decommitted by $\mathcal{A}$ when interacting with S.

Indistinguishability of the simulation. We start with the one-left many-right case and then we will consider the many-left many-right case. We consider a sequence of experiments $\mathsf{Expt}_i^{\mathcal{A}}(m, z)$ and show that any distinguisher D between the experiments can be used to produce a contradiction. Therefore, the output of each experiment is the output of a distinguisher D (which existence is assumed by contradiction) on input a message m and a vector $\tilde{M}$ whose i-th component $\tilde{m}_i$ is defined as follows. If the decommitment phase of the i-th right interaction terminates successfully and its transcript is different from all the left interactions,

then $\tilde{m}_i$ is the message that $\mathcal{A}$ has decommitted in the i-th right interaction. Otherwise, $\tilde{m}_i = \perp$. We also set $p_i^{\mathcal{A}}(m,z) = \text{Prob}[\ \mathsf{Expt}_i^{\mathcal{A}}(m,z) = 1\ ]$.

$\mathsf{Expt}_0^{\mathcal{A}}(m,z)$ is the experiment in which $\mathcal{A}$ plays with S that behaves as a honest receivers in the right interactions and as a honest committer on input m in the left interaction. We notice that, since S is acting as honest receiver and honest committer, $p_0^{\mathcal{A}}(m,z)$ is the probability that D outputs 1 on input distributed according to $\mathsf{cmim}_{\mathsf{Dec}}^{\mathcal{A}}(m,z)$.

Experiment $\mathsf{Expt}_1^{\mathcal{A}}(m,z)$ differs from Expt_0 only because in the left commitment phase, S runs the extractor of the WIPoK used by $\mathcal{A}$. Since there is no other deviation, we have that $p_1^{\mathcal{A}}(m,z) = p_0^{\mathcal{A}}(m,z)$.

Experiment $\mathsf{Expt}_2^{\mathcal{A}}(m,z)$ differs from Expt_1 in that in the left decommitment phase, S executes the code of the honest prover but uses a fake witness (that is the witness extracted in the left commitment phase from $\mathcal{A}$'s WIPoK). Next we prove that $|p_2^{\mathcal{A}}(m,z) - p_1^{\mathcal{A}}(m,z)|$ is negligible. Assume by contradiction that this difference is non-negligible; as the only difference between the two games consists in the witness used in the nmZK played in the decommitment phase, we show how to break the witness indistinguishability of nmZK. Specifically, we play the following game with an external prover P. We perform the commitment phase like in game $\mathsf{Expt}_1^{\mathcal{A}}(m,z)$. In particular, in the left commitment phase S has computed and sent to $\mathcal{A}$ commitment $c = \mathsf{Com}(m,s)$ and $\mathcal{A}$ has produced commitments c_0 and c_1 and proved knowledge of the message committed to by one of the two. We denote by (m_b, s_b) the witness extracted by S from $\mathcal{A}$'s WIPoK. The decommitment phase proceeds as in game Expt_1 with the exception of the execution of nmZK in the left decommitment phase which is performed by the external prover P. P is fed with the real witness (m,s) and the fake witness (m_b, s_b) and performs the code of the honest prover using one of the two. Notice that the decommitment phase is straight-line. We observe that if P uses the fake witness then we are actually playing game $\mathsf{Expt}_2^{\mathcal{A}}(m,z)$ whereas if P uses the real witness we are playing $\mathsf{Expt}_1^{\mathcal{A}}(m,z)$. Therefore if D distinguishes these two games, we break the witness indistinguishability of nmZK. We stress that in this reduction we have not used the extractor of the nmZK of the decommitment phase, therefore we can relay messages with P without rewinding it.

Next we consider $\mathsf{Expt}_3^{\mathcal{A}}(m,z)$ in which S uses the simulator of nmZK in the left commitment phase. Since the simulation is perfect we have that $p_3^{\mathcal{A}}(m,z) = p_2^{\mathcal{A}}(m,z)$.

Next we consider $\mathsf{Expt}_4^{\mathcal{A}}(m,z)$ in which S commits to 0^k in the left commitment phase. Any distinguisher between $\mathsf{Expt}_4^{\mathcal{A}}(m,z)$ and $\mathsf{Expt}_3^{\mathcal{A}}(m,z)$ can be easily reduced to a distinguisher between a commitment of 0^k and a commitment of m using Com, by simply playing this commitment as c, completing the experiment and then giving in output the same output of the distinguisher. Therefore by the computational hiding of Com we have that $|p_4^{\mathcal{A}}(m,z) - p_3^{\mathcal{A}}(m,z)|$ is negligible.

Next we consider $\mathsf{Expt}_5^{\mathcal{A}}(m,z)$ in which S runs the honest prover of nmZK in the left commitment phase. Since the simulation is perfect we have that $p_5^{\mathcal{A}}(m,z) = p_4^{\mathcal{A}}(m,z)$.

This sequence of experiments shows that the distribution of the messages decommitted by $\mathcal{A}$ during the man-in-the-middle game when the honest sender commits and decommits to m and $\mathcal{A}$ commits and decommits with the honest receiver R (i.e., $\mathsf{Expt}_0^{\mathcal{A}}(m,z)$), is indistinguishable from the distribution of the messages that $\mathcal{A}$ decommits in the simulated game where S plays both as sender committing to 0^k and as receiver (i.e., $\mathsf{Expt}_5^{\mathcal{A}}(m,z)$).

Epilogue. We now show that S is actually a stand-alone adversary, i.e., it can commit and open to a honest receiver R the same messages that $\mathcal{A}$ can open and decommit during a man-in-the-middle game.

Following the description of S, we know that S commits to R the messages that it extracts from $\mathcal{A}$ at the end of the commitment phase of the simulated game. The proof of non-malleability with respect to commitment given previously, says that the messages committed by S to R have the same distribution of the ones committed by $\mathcal{A}$ in the real game. Then the description of S says that S decommits to R the commitments that correspond to the ones that $\mathcal{A}$ decided to decommit to S in the decommitment phase of the simulated game. Since the indistinguishability of the simulation proved so far says that $\mathcal{A}$ decommits to S the same messages that $\mathcal{A}$ decommits in the real game, we have that S decommits to R the same messages decommitted by $\mathcal{A}$ in the real game, unless $\mathcal{A}$ in the real game decommits messages different with respect to the committed ones (indeed, S never decommits to R a message that is different from the committed one).

Therefore we now show that in the real game $\mathcal{A}$ can not open to different messages, this will imply that S decommits to R messages with the same distribution of the ones decommitted by $\mathcal{A}$.

In the real game $\mathcal{A}$ cannot open in a different way. Assume by contradiction that, with some non-negligible probability, in the real game (i.e., when $\mathcal{A}$ plays with a honest prover committing to m and with honest receivers) there exists i such that the decommitted message m_i' is different from the committed message m_i[5]. We denote by c_0 and c_1 the two commitments computed by R in the i-th commitment phase of $\mathcal{A}$ and by $b \in \{0,1\}$ the bit such that the receiver R used knowledge of the message committed to by c_b to perform the WIPoK of the i-th commitment phase. Given that $\mathcal{A}$ successfully completes the i-th decommitment phase then, we can consider the following experiment. Adversary $\mathcal{A}$ plays with a real sender and a receiver-extractor. The real sender commits to m, while the receiver-extractor runs the honest receiver algorithm for all right commitments and runs the extractor of nmZK of the i-th decommitment phase. The receiver-extractor with overwhelming probability outputs a pair $(\hat{m},\hat{s})$ such that either $c_b = \mathsf{Com}(\hat{m},\hat{s})$ or $c_{1-b} = \mathsf{Com}(\hat{m},\hat{s})$ (i.e., since $\mathcal{A}$ decommitted to a different message, the witness must be a fake one).

Suppose that with some non-negligible probability it happens that $c_{1-b} = \mathsf{Com}(\hat{m},\hat{s})$. Then we break the hiding property of Com. Consider the following

[5] The committed message is the one uniquely specified by the statistically binding commitment scheme used as sub-protocol.

adversary $\mathcal{B}$ that receives a commitment $\hat{c}$ and would like to compute the message committed to by $\hat{c}$ with some non-negligible probability. $\mathcal{B}$ interacts with $\mathcal{A}$ and plays all commitment phases as the honest senders and receivers, with the only exception of the i-th commitment phase played as receiver. Here $\mathcal{B}$ picks a random $b \in \{0,1\}$, a random $m_b \in \{0,1\}^k$ and random $s_b \in \{0,1\}^k$ and computes commitment $c_b = \mathsf{Com}(m_b, s_b)$ and sets $c_{1-b} = \hat{c}$. Then $\mathcal{B}$ continues the commitment phase by running the code of the honest prover wiP of the WIPoK using (m_b, s_b) as witness. By our hypothesis, with some non-negligible probability, the extractor gives the message committed to by $\hat{c}$, this gives to $\mathcal{B}$ a non-negligible advantage for breaking the hiding property of Com.

Suppose instead that, except with negligible probability, it happens that $c_b = \mathsf{Com}(\hat{m}, \hat{s})$. We show that the witness indistinguishability of the WIPoK is violated. More specifically, we consider a WI adversary $\mathcal{B}$ that executes internally all the previous interactions with the only exception that the WIPoK of the i-th right commitment phase is played by relaying messages with an external prover (that uses a witness for $c_{b^\star}$ for some $b^\star \in \{0,1\}$). $\mathcal{B}$ then plays internally the decommitment phases with the exception of the i-th decommitment phase for which the extractor is used. By looking at the extracted witness, $\mathcal{B}$ will guess the witness used by the external prover.

Summing up. We have therefore shown that $\mathcal{A}$ decommits successfully only the committed messages. Moreover, we have shown that in the simulated game $\mathcal{A}$'s choices for which commitment have to be decommitted are indistinguishable from its choices in the simulated game. These two properties guarantee that S decommits to R messages indistinguishable from the ones decommitted by $\mathcal{A}$ in the real game.

This terminates the proof for the one-left many-right case.

The many-left many-right case for concurrent NMd. Let $l = \mathsf{poly}(k)$ be the size of the vector of messages M, we consider the hybrid games $\{\mathsf{Expt}_i^{\mathcal{A}}\}_{0 \le i \le l}$, where $\mathsf{Expt}_i^{\mathcal{A}}$ for $i = 0, \ldots l$ is defined as follows. In the game $\mathsf{Expt}_i^{\mathcal{A}}$ the committer commits to m_j as the j-th commitments if $j \le i$, and to 0^k if $j > i$. Moreover in $\mathsf{Expt}_i^{\mathcal{A}}$ the i-th commitment is decommitted using a legal witness if $j \le i$ and a fake witness if $j > i$. Obviously $\mathsf{Expt}_0^{\mathcal{A}}$ corresponds to the game played by the simulator (including both the commitment and decommitment phases) while $\mathsf{Expt}_l^{\mathcal{A}}$ corresponds to game played by the honest committer (again, including both the commitment and decommitment phases). For all M and z we denote by $\{\mathsf{csis}_{\mathsf{Dec}}^{\mathsf{Expt}_i^{\mathcal{A}}}(M, z)\}$ the random variable that associates to each successfully completed decommitment phase of $\mathsf{Expt}_i^{\mathcal{A}}$ the messages decommitted by $\mathcal{A}$. Instead $\{\mathsf{csis}_{\mathsf{Dec}}^{\mathsf{Expt}_i^{\mathcal{A}}}(M, z)\}$ associates the value $\perp$ to interactions that have not been completed by $\mathcal{A}$.

Assume by contradiction that the scheme is not concurrent non-malleable with respect to decommitment. It follows that there must be an index $i \in \{0, \ldots l-1\}$ such that D distinguishes with non-negligible probability between $\{\mathsf{csis}_{\mathsf{Dec}}^{\mathsf{Expt}_i^{\mathcal{A}}}(M, z)\}$ and $\{\mathsf{csis}^{\mathsf{Expt}_{i+1}^{\mathcal{A}}}\mathsf{Dec}(M, z)\}$. The only difference between game

$\mathsf{Expt}_i^{\mathcal{A}}$ and game $\mathsf{Expt}_{i+1}^{\mathcal{A}}$ for $i \in \{0, \ldots l-1\}$ is that the $i+1$ commitment is computed for message 0^k in $\mathsf{Expt}_i^{\mathcal{A}}$ while it is computed for message m_i in $\mathsf{Expt}_{i+1}^{\mathcal{A}}$. Moreover the corresponding decommitment uses a fake witness in $\mathsf{Expt}_i^{\mathcal{A}}$ and a legal witness in $\mathsf{Expt}_{i+1}^{\mathcal{A}}$.

We can therefore construct a successful MiM adversary $\mathcal{A}'$ for the one-left many-right case. Adversary $\mathcal{A}'$ internally runs all left sessions with the only exception of the $(i+1)$-st commitment and the corresponding decommitment that is played either with a honest committer committing to m_{i+1} or with the simulator of the one-left many-right case. Therefore $\mathcal{A}'$ breaks the one-left many-right non-malleability which is a contradiction.

From the previous discussion and by observing that existence of a family of claw-free permutations is sufficient for the tools we use, we have the following theorem and corollary.

Theorem 2. *Under the assumption of existence of a tag-based one-left many-right perfect cNMZK arguments of knowledge for all* NP*, of a secure signature scheme and of a statistically-binding non-interactive commitment scheme, commitment scheme $\mathcal{NM}cd$ is both simulation-based concurrent NMc and simulation-based concurrent NMd.*

Corollary 1. *Under the existence of a family of claw-free permutations there exists a constant-round commitment scheme that is both simulation-based concurrent NMc and simulation-based concurrent NMd.*

4 Acknowledgments

We thank the anonymous reviewers for their suggestions. The work of the first author has been supported in part by IBM Faculty Award, Xerox Innovation Group Award, NSF grants 0430254, 0716835, 0716389, 0830803 and U.C. MICRO grant. The work of the authors has been supported in part by the European Commission through the EU IST program under Contract IST-2002-507932 ECRYPT, and the one of the last two authors through the the EU ICT program under Contract ICT-2007-216646 ECRYPT II and through the FP6 program under contract FP6-1596 AEOLUS.

References

1. Di Crescenzo, G., Ishai, Y., Ostrovsky, R.: Non-interactive and non-malleable commitment. In: 30th Annual ACM Symposium on Theory of Computing, Dallas, Texas, USA, ACM Press (1998) 141–150
2. Di Crescenzo, G., Katz, J., Ostrovsky, R., Smith, A.: Efficient and non-interactive non-malleable commitment. In Pfitzmann, B., ed.: Advances in Cryptology – EUROCRYPT 2001. Volume 2045 of Lecture Notes in Computer Science., Innsbruck, Austria, Springer-Verlag, Berlin, Germany (2001) 40–59

3. Pass, R., Rosen, A.: New and Improved Constructions of Non-Malleable Cryptographic Protocols. In: 37th Annual ACM Symposium on Theory of Computing, ACM Press (2005) 533–542
4. Pass, R., Rosen, A.: Concurrent non-malleable commitments. In: 46th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press (2005) 563–572
5. Blum, M.: Coin flipping by telephone. In: Proc. IEEE Spring COMPCOM. (1982) 133–137
6. Goldreich, O., Micali, S., Wigderson, A.: Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. In: 27th Annual Symposium on Foundations of Computer Science, Toronto, Ontario, Canada, IEEE Computer Society Press (1986) 174–187
7. Dolev, D., Dwork, C., Naor, M.: Non-malleable cryptography. In: 23rd Annual ACM Symposium on Theory of Computing, New Orleans, Louisiana, USA, ACM Press (1991) 542–552
8. Barak, B.: Constant-round coin-tossing with a man in the middle or realizing the shared random string model. In: 43rd Annual Symposium on Foundations of Computer Science, Vancouver, British Columbia, Canada, IEEE Computer Society Press (2002) 345–355
9. Pass, R., Rosen, A.: Concurrent nonmalleable commitments. SIAM Journal on Computing **37** (2008) 1891–1925
10. Feige, U.: Alternative Models for Zero Knowledge Interactive Proofs. Weizmann Institute of Science (1990)
11. Dwork, C., Naor, M., Sahai, A.: Concurrent zero-knowledge. In: 30th Annual ACM Symposium on Theory of Computing, Dallas, Texas, USA, ACM Press (1998) 409–418
12. Ostrovsky, R., Persiano, G., Visconti, I.: Constant-round concurrent non-malleable zero knowledge in the bare public-key model. In: ICALP. Volume 5126 of LNCS., Springer-Verlag (2008) 548–559
13. Ostrovsky, R., Persiano, G., Visconti, I.: Concurrent non-malleable witness indistinguishability and its applications. Technical Report ECCC Report TR06-095, ECCC (2006)
14. Barak, B., Prabhakaran, M., Sahai, A.: Concurrent non-malleable zero knowledge. In: 47th Annual Symposium on Foundations of Computer Science, IEEE Computer Society Press (2006)
15. Ostrovsky, R., Persiano, G., Visconti, I.: Constant-round concurrent non-malleable commitments and decommitments. Technical Report 2008/235, Cryptology ePrint Archive (2008)
16. Blum, M.: How to Prove a Theorem So No One Else Can Claim It. In: Proceedings of the International Congress of Mathematicians. (1986) 1444–1451
17. Feige, U., Shamir, A.: Witness indistinguishable and witness hiding protocols. In: 22nd Annual ACM Symposium on Theory of Computing, Baltimore, Maryland, USA, ACM Press (1990) 416–426
18. Feige, U., Lapidot, D., Shamir, A.: Multiple NonInteractive Zero Knowledge Proofs under General Assumptions. SIAM Journal on Computing **29** (1999) 1–28