Improved Black-Box Constructions of Composable Secure Computation

Rohit Chatterjee
Stony Brook University, NY, USA
rochatterjee@cs.stonybrook.edu

Xiao Liang
Stony Brook University, NY, USA
liang1@cs.stonybrook.edu

Omkant Pandey
Stony Brook University, NY, USA
omkant@cs.stonybrook.edu

Abstract

We close the gap between black-box and non-black-box constructions of *composable* secure multiparty computation in the plain model under the *minimal assumption* of semi-honest oblivious transfer. The notion of protocol composition we target is *angel-based* security, or more precisely, security with super-polynomial helpers. In this notion, both the simulator and the adversary are given access to an oracle called an *angel* that can perform some predefined super-polynomial time task. Angel-based security maintains the attractive properties of the universal composition framework while providing meaningful security guarantees in complex environments without having to trust anyone.

Angel-based security can be achieved using non-black-box constructions in $\max(R_{\mathsf{OT}}, \widetilde{O}(\log n))$ rounds where R_{OT} is the round-complexity of semi-honest oblivious transfer. However, current best known *black-box* constructions under the same assumption require $\max(R_{\mathsf{OT}}, \widetilde{O}(\log^2 n))$ rounds. If R_{OT} is a constant, the gap between non-black-box and black-box constructions can be a multiplicative factor $\log n$. We close this gap by presenting a $\max(R_{\mathsf{OT}}, \widetilde{O}(\log n))$ round black-box construction. We achieve this result by constructing constant-round 1-1 CCA-secure commitments assuming only black-box access to one-way functions.

2012 ACM Subject Classification Security and privacy → Mathematical foundations of cryptography

Keywords and phrases Secure Multi-Party Computation, Black-Box, Composable, Non-Malleable

Digital Object Identifier 10.4230/LIPIcs.ICALP.2020.28

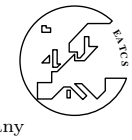
Category Track A: Algorithms, Complexity and Games

Related Version A full version of the paper is available at `https://eprint.iacr.org/2020/494`.

Funding This material is based upon work supported in part by DARPA SIEVE Award HR00112020026, NSF grant 1907908, and a Cisco Research Award. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Government, DARPA, NSF, or Cisco.

1 Introduction

Secure multiparty computation (MPC) [79, 24] enables two or more mutually distrustful parties to compute any functionality without compromising the privacy of their inputs. These early results [79, 24], along with a rich body of followup work that refined and developed the concept [25, 5, 65, 7, 73, 8], demonstrated the feasibility of general secure computation and its significance to secure protocol design. The existence of semi-honest oblivious transfer (OT) was established by Kilian [49] as the minimal, i.e., necessary and sufficient, assumption for general secure computation. The focus of this work is on *black-box constructions* of *composable* MPC protocols. We discuss these two aspects.

47th International Colloquium on Automata, Languages, and Programming (ICALP 2020).
Editors: Artur Czumaj, Anuj Dawar, and Emanuela Merelli; Article No. 28; pp. 28:1–28:20


Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Black-Box constructions. A construction is black-box if it does not refer to the code of any cryptographic primitive it uses, and only depends on their input/output behavior. Such constructions are usually preferable since their efficiency is not affected by the implementation details of the underlying cryptographic primitives; moreover, they remain valid and applicable if the code of the underlying primitives is simply not available, e.g., in case of constructions based on hardware tokens [64, 23, 46, 28, 38].

Early constructions of general-purpose MPC were non-black-box in nature particularly due to NP-reductions required by underlying zero-knowledge proofs [24]. Ishai et al. [41] presented the first black-box construction of general purpose MPC based on enhanced trapdoor permutations or homomorphic public-key encryption schemes. Together with the subsequent work of Haitner [36], this provided a black-box construction of a general MPC protocol under minimal assumptions (i.e., semi-honest OT). The round complexity of black-box MPC was improved to $O(\log^* n)$ rounds by Wee [78], and to constant rounds by Goyal [26]. In the two party setting, a constant round construction was first obtained by Pass and Wee [72], and subsequently a 5-round construction was given by Ostrovsky, Richelson, and Scafuro [68], which is known to be optimal by the results of Katz and Ostrovsky [47].

Composable security. The notion of security considered in early MPC works is called *standalone security* since it only considers a single execution of the protocol. Stronger notions of security are required for complex environments such as the Internet where several MPC protocols may run *concurrently*. This setting is often referred to as the *concurrent* setting, and unfortunately, as shown by Feige and Shamir [19], stand-alone security does not necessarily imply security in the concurrent setting.

To address this issue, Canetti [8] proposed the notion of *universally composable* (UC) security which has two important properties: *concurrent security* and *modular analysis*. The former means that UC secure protocols maintain their security in the presence of other *concurrent* protocols and the latter means that the security of a larger protocol in the UC framework can be derived from the UC security of its component protocols. This latter property is stated as a composition theorem which, roughly speaking, states that UC is closed under composition [8]. Unfortunately, UC security turns out to be impossible in the plain model for most tasks [8, 9, 11]. Relaxations of UC that consider composing the same protocol were also ruled out by Lindell [59, 60].

These strong negative results motivated the search for alternative notions of concurrent security in the plain model by endowing more power to the simulator such as super-polynomial resources [69, 75, 6], ability to receive multiple outputs [30, 29], or resorting to weaker notions such as bounded concurrency [1, 70], input indistinguishability [63], or a combination thereof [27]. While all of these notions were (eventually) achieved under polynomial hardness assumptions [75, 4, 62, 12, 21, 55, 71, 52, 50, 32, 6, 22], only angel-based security by Prabhakaran and Sahai [75] (including its extension to interactive angels by Canetti, Lin, and Pass [12]) and shielded-oracle security by Broadnax et al. [6] are known to have the modular analysis property, i.e., admitting a composition theorem along the lines of UC. We focus on angel-based security in this work since it arguably has somewhat better composition properties than shielded oracles.[1]

[1] As noted in [6], shielded oracle security does not technically have the modular analysis property and is actually strictly weaker than angel-based. Nevertheless, it is still "compatible" with the UC framework – the security of a composed protocol can be derived from that of its components.

Angel based security is similar to UC except that it allows the simulator as well as the adversary access to a super-polynomial resource called an "angel" which can perform a pre-defined task such as inverting a one-way function. Early constructions of angel-based security were based on non-standard assumptions [75, 4, 62]. The beautiful work of Canetti et al. [12] presented the first construction under polynomial hardness assumptions, and the subsequent work of Goyal et al. [32] improved the round complexity to $\widetilde{O}(\log \lambda)$ under general assumptions.

The first *black-box construction* of angel-based security was obtained by Lin and Pass [55], under the minimal assumption of semi-honest OT. The main drawback of [55] is that it requires *polynomially many* rounds even if the underlying OT protocol has constant rounds. To address this situation, Kiyoshima [50] presented a $\widetilde{O}(\log^2 \lambda)$-round construction assuming constant-round semi-honest OT (or alternatively, $\max(\widetilde{O}(\log^2 \lambda), O(\mathrm{R}_{\mathsf{OT}}))$ rounds where R_{OT} is OT's round-complexity). We remark that Broadnax et al. [6] present a constant-round black-box construction for (the weaker but still composable) shielded-oracle security (utilizing prior work by Hazay and Venkitasubramaniam [39] who provide a constant-round protocol in the CRS-hybrid model); however, they require stronger assumptions, specifically, homomorphic commitments and public-key encryption with oblivious public-key generation.

State of the art. To summarize our discussion above, under the *minimal assumption* of polynomially secure semi-honest OT, the best known round complexity of *black-box constructions* for angel-based security, and in fact any composable notion with modular analysis property, is due to Kiyoshima [50] which requires $\max(\widetilde{O}(\log^2 \lambda), O(\mathrm{R}_{\mathsf{OT}}))$ rounds. This is in contrast to the non-black-box construction of Goyal et al. [32] which requires only $\max(\widetilde{O}(\log \lambda), O(\mathrm{R}_{\mathsf{OT}}))$ rounds. Therefore, there is a multiplicative gap of $\widetilde{O}(\log \lambda)$ between the round-complexities of state-of-the-art black-box and non-black-box constructions of angel-based MPC if, e.g., semi-honest OT has at most logarithmic rounds.

1.1 Our Results

In this work, we prove the following theorem, thus closing the gap between the round complexity of black-box and non-black-box constructions of angel-based MPC under minimal assumptions:

▶ **Theorem 1** (Main). *Assume the existence of* R_{OT}*-round semi-honest oblivious transfer protocols. Then, there exists a* $\max(\widetilde{O}(\log \lambda), O(\mathrm{R}_{\mathsf{OT}}))$*-round black-box construction of a general MPC protocol that satisfies angel-based UC security in the plain model.*

Note that this yields a $\widetilde{O}(\log \lambda)$-round construction under the general assumption of enhanced trapdoor permutations since they imply constant-round semi-honest OT.

We follow the framework of [12] and its extensions in [55, 50]. The main building block [12] is a special commitment scheme called a CCA-Secure Commitment. Roughly speaking, a CCA-secure commitment is a tag-based commitment scheme that maintains hiding even in the presence of a decommitment oracle $\mathcal{O}$. More specifically, the adversary receives one commitment from an honest committer and may simultaneously make concurrently many commitments to $\mathcal{O}$ (similar to non-malleable commitments [17]). The oracle immediately extracts and sends back any value adversary commits successfully provided that it used a tag that is different from the one used by the honest committer. Lin and Pass [55] show that $O(\max(\mathrm{R}_{\mathsf{CCA}}, \mathrm{R}_{\mathsf{OT}}))$-round black-box angel-based MPC can be obtained from a $\mathrm{R}_{\mathsf{CCA}}$-round CCA commitment and a R_{OT}-round semi-honest OT protocol. Kiyoshima [50] demonstrated that $\widetilde{O}(k \cdot \log \lambda)$-round CCA-secure commitments can be obtained in a black-box manner

from a k-round commitment scheme with slightly weaker security called "one-one CCA" where the adversary can open only one session each with the committer as well as the oracle; they further construct a $O(\log \lambda)$-round one-one CCA scheme from one-way functions in a black-box manner.

We instead present a *constant round* construction of one-one CCA, which implies $\widetilde{O}(\log \lambda)$-round (full) CCA commitments using [50] (and Theorem 1 using [55]):

▶ **Theorem 2** (CCA Secure Commitments). *Assume the existence of one-way functions. Then, there exists a $\widetilde{O}(\log \lambda)$-round black-box construction of a CCA-secure commitment scheme.*

1.2 Overview of Techniques

Current approaches. Let us briefly review the current approaches for constructing CCA secure commitments. The main difficulty in constructing CCA secure commitments under polynomial hardness is to move from the real world – which contains the exponential time decommitment oracle $\mathcal{O}$– to a hybrid where $\mathcal{O}$'s responses can be efficiently simulated. A standard way to do this is to use a argument-of-knowledge (AoK): the protocol should require the (man-in-the-middle) adversary, say $\mathcal{A}$, to give a AoK of the value it commits. The main difficulty in employing this is that $\mathcal{A}$ may open concurrently many sessions with $\mathcal{O}$ (referred here to as "right" side sessions), interleaved in an arbitrary manner; furthermore, these values have to be extracted *immediately* within each session irrespective of what happens in other sessions. This is precisely the issue in constructing (black-box simulatable) concurrent zero-knowledge (CZK) protocols [18] as well, and ideas from there are applied in this setting too. A second difficulty is that these extractions must happen without rewinding the commitment $\mathcal{A}$ *receives* (referred to as "left" side session).

It is worthwhile to quickly recall the (tag based) non-malleable commitment construction in the original work of [17]. In this construction, $\mathcal{A}$ has only one right session; to prove that the value on the right is (computationally) independent from that on the left, the value on the right is extracted without rewinding the sensitive parts of the left side commitments. This is done by creating two types of AoK– one each for two possible values of a bit. These AoK create rewinding "slots" for extraction such that if $\mathcal{A}$ uses a different bit in the tag, it risks the possibility of having to perform a AoK on its own – i.e., without any "dangerous" rewinding on the left – in one of the slots (called a "free" slot). These special AoK are performed for each bit of the tag *sequentially* so that at least one free slot is guaranteed since the left and right tags are different by definition. While this requires n rounds n-bit tags, it is possible to split the tag into n smaller tags of $\log n$ bits and run the protocol for each of them in parallel [17, 57]. Referred to as "LOG trick," this yields a $O(\log n)$-round protocol.

The key idea for CCA commitments in [12], at a high level, is to ensure that in the *concurrent* setting, many free slots exist for each session so that extraction succeeds before the end of that session. This is achieved by creating a polynomial round protocol consisting of sequential repetition of special AoK as above and then relying on an analysis that is, at a high level, similar to early rewinding techniques from CZK literature [76, 10]. Once the issue of concurrent extraction is handled, the additional ideas in [55] are (again, at a high level) to enforce this approach using cut-and-choose protocols to obtain a black-box construction. The work of Goyal et al. [32] shows how to separate the tasks of "concurrent extraction" and"non-malleability" in this approach by proving a "robust extraction lemma." This allows them to follow a structure similar to that of concurrent non-malleable zero-knowledge (CNMZK) from [3] which matches the round complexity of CZK, i.e., $\widetilde{O}(\log n)$. However, their approach requires non-black usage of one-way functions. Kiyoshima [50] shows that the

robust-extraction lemma can actually be applied to the previous black-box protocol of [55] to get $\widetilde{O}(k \cdot \log n)$ rounds if one has a slightly stronger primitive than non-malleable commitments: namely k-round 1-1 CCA commitments. To build such commitments, Kiyoshima builds non-malleability "from scratch" by combining the DDN "LOG trick" with cut-and-choose components of [55] so that the extraction on right in the standalone setting, can be done without any dangerous rewinding on left. This however results in $O(\log n)$ rounds for 1-1 CCA and $\widetilde{O}(\log^2 n)$ for full CCA.

Our approach. We significantly deviate from current approaches for constructing 1-1 CCA commitments. Instead of attempting to build non-malleability *from scratch*, our goal is to have a generic construction built around existing non-malleable commitments. The resulting protocol will not only have a simpler and more modular proof of security, but will also benefit from the efficiency and assumptions of the underlying non-malleable commitment (NMCom). Towards this goal, we return to investigate the structure of CNMZK protocols even for the simpler case of 1-1 CCA.

Setting aside the issue of round-complexity for the moment, a key idea in the construction of CNMZK protocols [3, 56, 67, 54] is to have the prover give a non-malleable commitment (NMCom) which can later be switched to a "trapdoor value" set by the verifier; the non-malleability of NMCom ensures that $\mathcal{A}$ cannot switch his value to a trapdoor on the right (unless he did so in the real world, which can be shown to be impossible through other means). The prover later proves that either the statement is true or it committed the trapdoor. The main problem with this approach is that it requires us to prove a predicate over the value committed in NMCom which requires non-black-box use of cryptographic primitives.

Non-malleable commit-and-prove. One potential idea to avoid non-black-box techniques is to turn to black-box commit-and-prove protocols in the literature and try to re-develop them in the context of non-malleability. Commit-and-prove protocols allow a committer to commit to a value v so that later, it can prove a predicate ϕ over the committed value in zero-knowledge. These protocols can be constructed in constant rounds using the powerful "MPC-in-the-head" approach introduced by Ishai et al. [42]. The approach allows committing multiple values $v_1, \ldots, v_n$ and then proving a joint predicate ϕ over them. One such construction is implicit in the work of Goyal et al. [31]. Such commitments were also used extensively by Goyal et al. to build size-hiding commit-and-prove [33] and an optimal four round construction was obtained by Khurana, Ostrovsky, and Srinivasan [48]. As noted above, if we can develop an appropriate non-malleable version of such protocols, it is conceivable that they can yield constant-round 1-1 CCA commitment. However, that non-malleable commitments are not usually equipped to handle proofs. Thus, such an approach will necessarily have to "open up" the construction of non-malleable commitments. In particular, like previous constructions, this approach cannot be based on non-malleable commitments in a black-box manner.

Changing the direction of NMCcom. In order to rely on non-malleable commitments directly, it is essential that we do not prove anything about the values committed inside the NMCom. Instead, we should restrict all proofs to be performed only over standard commitments since for them we can use standard black-box commit-and-prove protocols. Towards building this property, what if we change the direction of NMCom and ask the receiver of 1-1 CCA to send non-malleable commitments, which, for example, can be opened later? More specifically, in our 1-1 CCA protocol, the receiver will send a NMCom to a random value σ which it will open subsequently. The committer will send a "trapdoor

commitment" t before it sees σ opened. Later, the committer will commit to the desired value v and give a AoK that either it knows v or t is a commitment to σ (the "trapdoor"). Observe that this structure completely avoids any proof directly over non-malleable commitments; all proofs only need to be performed over ordinary commitments. Therefore, if we use the commit phase of black-box commit-and-prove protocols to commit to σ and v we can easily complete the AoK in a black-box manner: the predicate ϕ in the proof phase will simply test for the presence of trapdoor σ. Some standard soundness issues arise in this approach but they can be handled by ensuring that the commit phase is extractable.

Although this approach yields a black-box construction directly from NMCom, it is hard to prove the 1-1 CCA property. At a high level, this is because of the following: if in the 1-1 CCA game, $\mathcal{A}$ schedules the completion of the left NMCom *before* the right one[2], the simulator in the security proof must extract σ from this NMCom while the right NMCom is still in play (so that it can generate t to be a commitment to σ). This involves rewinding the left NMCom (assuming it is extractable) which in turn rewinds the right session.[3] A similar issue arises in the work of Jain and Pandey [44] on black-box non-malleable zero-knowledge where it is resolved by using a NMCom that is already 1-1 CCA secure. We do not have this flexibility in our setting.

A possible fix for this issue is to rely on some kind of "delayed input" property: i.e., the commitment to t will be an extractable commitment that does not require the message m to be committed until the last round. This property can be obtained by committing to a key k in an extractable manner and then in the last round committing to m by simply encrypting with k. This however will no longer be compatible with the black-box commit-and-prove strategy since we will now have to take encryption into account.

We overcome this issue by making extensive use of extractable commitments. More specifically, we first *prepend* the NMCom with a standard "slot-based" extractable commitment which commits to the same value σ as the NMCom. If the NMCom also has a slot like extractable structure (e.g., the *three round* scheme of [34]), we can argue that non-synchronous adversaries must always leave a free slot either on top or at the bottom of NMCom. For example, in the troublesome scheduling discussed above, $\mathcal{A}$ can be easily rewound in the last two messages of NMCom (if we use [34]) *without* rewinding the right NMCom. In other non-synchronous schedules it will have a free slot in the top extractable commitment on the left. On the other hand, synchronous adversaries will fail in the NMCom step (and synchronous non-malleability suffices for our purposes). In summary, this will suffice for us to show that even if our simulator sets up the trapdoor statement on the left (by committing σ in t), $\mathcal{A}$ cannot do the same on the right. Other NMCom, particularly public-coin extractable NMCom also seem sufficient.

A second issue here is the intertwining of the left AoK[4] with "extractable" components on the right, e.g., the right AoK (or extractable commitment steps). In order to prove that $\mathcal{A}$ cannot setup the trapdoor, extraction from right AoK will be necessary and this will be troublesome when changing the witness in the left AoK during hybrids. This issue can be handled using the sequential repetition technique from [53]: we use $k+1$ AoKs where k is the (constant) rounds in a single AoK. Also note that other common methods for handling this issue do not work: e.g., we cannot rely on *statistical* WI since it requires stronger assumptions

[2] Note that NMCom's direction is opposite to that of 1-1 CCA: the receiver of 1-1 CCA is the sender of right NMCom.

[3] This is not an issue in the synchronous schedule since in that case, the value $\mathcal{A}$ commits to in NMCom is provided to the distinguisher along with the joint view.

[4] Observe that the AoK will just be the proof part of appropriate black-box commit-and-prove with right parameters to ensure black-box property; they will also satisfy witness-indistinguishability [19].

for constant rounds; we also cannot use proofs that are secure against a fixed number of rewinds since they usually allow a noticeable probability of extraction which is insufficient for 1-1 CCA commitments, where extraction must succeed with overwhelming probability.

1.3 Other Related Works

The focus of our work is constructions in the plain model. Hazay and Venkitasubramaniam [40] gave a black-box construction of an MPC protocol without any setup assumptions that achieves composable security against an *adaptive* adversary. UC security can be achieved by moving to other trusted setup models such as the common reference string model [14, 9, 35], assuming an honest majority of parties [11], trusted hardware [64, 23, 46, 15], timing assumptions on the network [45], registered public-key model [2], setups that may be expressed as a hybrid of two or more of these setups [20], and so on. Lin, Pass, and Venkitasubramaniam [58, 71] show that a large number of these setup models could be treated in a unified manner, and black-box analogues of these results were obtained by Kiyoshima, Lin, and Venkitasubramaniam [51].

2 Preliminaries

Notation. We use λ for the security parameter. We use $\overset{c}{\approx}$ to denote computational indistinguishability between two distributions. For a set S, we use $x \xleftarrow{\$} \mathsf{S}$ to mean x is sampled uniformly at random from S. PPT denotes probabilistic polynomial time and $\mathsf{negl}(\cdot)$ denotes negligible function.

We assume familiarity with standard concepts such as commitment schemes, witness indistinguishability. In the following, we recall the definitions for extractable commitments, non-malleable commitments and CCA commitments. Definitions for the more basic primitives and other constructs (such as MPC related definitions) can be found in the full version of the paper [16].

2.1 Extractable Commitments

▶ **Definition 3** (Extractable Commitment Schemes). *A commitment scheme* $\mathsf{ExtCom} = (S, R)$ *is extractable if there exists an expected polynomial-time probabilistic oracle machine (the extractor)* Ext *that given oracle access to any PPT cheating sender* S^* *outputs a pair* (τ, σ^*) *such that:*

- ***Simulation:*** *τ is identically distributed to the view of S^* at the end of interacting with an honest receiver R in commitment phase.*
- ***Extraction:*** *the probability that τ is accepting and $\sigma^* = \perp$ is negligible.*
- ***Binding:*** *if $\sigma^* \neq \perp$, then it is statistically impossible to open τ to any value other than σ^*.*

The above construction of ExtCom (Figure 1) is standard [17, 74, 77, 70]. We will refer to it as the *standard* ExtCom.

▶ Remark 4 (Regarding Over-Extraction). Intuitively, Definition 3 stipulates that if the committer indeed commits to some value, the extractor must be able to extract it. We remark that it does not rule out what is called the "over-extraction" issue – namely, the extractor may extract a valid looking value even though none actually exists.

However, this definition suffices for most ZK and MPC applications (e.g. [75, 70, 72]), including ours. In our arguments specifically, we will employ hybrids where we extract the value that the adversary commits to using these commitment schemes. Jumping ahead, in the security proof for our protocol, we will need successful extraction only if the adversary actually commits to some *valid* value; otherwise, completing the hybrids is trivial.

The extractable commitment scheme, based on any commitment scheme Com, works in the following way. The scheme has 3 rounds if Com is non-interactive.

Input:

- Both S and R get security parameter 1^λ as the common input.
- S gets a string σ as his private input.

Commitmment Phase:

- The sender (committer) S commits using Com to λ pairs of strings $\{(v_i^0, v_i^1)\}_{i=1}^{\lambda}$ where $(v_i^0, v_i^1) = (\eta_i, \sigma \oplus \eta_i)$ and η_i are random strings in $\{0,1\}^{\ell(\lambda)}$ for $1 \leq i \leq \lambda$.[a]
- Upon receiving a challenge $\overline{c} = (c_1, \ldots, c_\lambda)$ from the receiver R, S opens the commitments to $(v_1^{c_1}, \ldots, v_\lambda^{c_\lambda})$.
- R checks that the openings are valid.

Decommitment Phase:

- S sends σ and opens the commitments to all λ pairs of strings.
- R checks that all the openings are valid, and also that $\sigma = v_1^0 \oplus v_1^1 = \cdots = v_\lambda^0 \oplus v_\lambda^1$.

[a] The scheme supports extraction as long as $k = \omega(\log \lambda)$ pairs are used.

Figure 1 Extractable Commitment Scheme $\langle S, R \rangle$.

2.2 Non-Malleable Commitments

We follow the definition of non-malleability from [57, 34]. This definition is based on the comparison between a real execution with an ideal one. In the *real* interaction, we consider a man-in-the-middle adversary $\mathcal{A}$ interacting with a committer C in the *left* session, and a receiver R in the *right*. We denote the relevant entities used in the right interaction as "tilde'd" version of the corresponding entities on the left. In particular, suppose that C commits to v in the left interaction, and $\mathcal{A}$ commits to $\widetilde{v}$ on the right. Let MIM_v denote the random variable that is the pair $(\mathsf{view}, \widetilde{v})$, consisting of the adversary's entire view of the man-in-the-middle execution as well as the value committed to by $\mathcal{A}$ on the right (assuming C commits to v on the left). The *ideal* interaction is similar, except that C commits to some arbitrary fixed value (say 0) on the left. Let MIM_0 denote the pair $(\mathsf{view}, \widetilde{v})$ in the ideal interaction. We use a *tag-based* (or "identity-based") specification, and ensure that $\mathcal{A}$ uses a distinct tag $\widetilde{\mathsf{id}}$ on the right from the tag id it uses on the left. This is done by stipulating that MIM_v and MIM_0 both output a special value $\perp_{\mathsf{id}}$ when $\mathcal{A}$ uses the same tag in both the left and right executions. The reasoning is that this corresponds to the uninteresting case when $\mathcal{A}$ is simply acting as a channel, forwarding messages from C on the left to R on the right and vice versa. We let $\mathsf{MIM}_v(z)$ and $\mathsf{MIM}_0(z)$ denote the real and ideal interactions respectively when the adversary receives auxiliary input z.

▶ **Definition 5** (Non-Malleable Commitment Schemes). *A (tag-based) commitment scheme $\langle C, R \rangle$ is non-malleable if for every* PPT *man-in-the-middle adversary $\mathcal{A}$, and for all values v, we have $\{\mathsf{MIM}_v(z)\}_{z \in \{0,1\}^*} \overset{c}{\approx} \{\mathsf{MIM}_0(z)\}_{z \in \{0,1\}^*}$.*

Synchronizing Adversaries. This notion refers to man-in-the-middle adversaries who upon receiving a message in one session, immediately respond with the corresponding message in the other session. An adversary is said to be *non-synchronizing* if it is not synchronizing.

2.3 CCA Commitments

We define the notion of CCA-secure commitments (and 1-1 CCA security in particular). These definitions rely on the notion of a *decommitment oracle*, which provide decommitments given valid transcripts to a particular (tag based) commitment protocol. Specifically, a decommitment oracle $\mathcal{O}$ for a given commitment protocol acts as follows:

- $\mathcal{O}$ acts as an honest reciever against some committer C, participating faithfully according to the specified commitment scheme. C is allowed to pick a tag for this interaction adaptively.
- At the end of this interaction, if the honest reciever were to accept the transcript as containing a valid commitment with respect to the given tag, $\mathcal{O}$ returns the value v committed by C to it. Otherwise, it returns $\perp$.

We denote an adversary with access to the decommitent oracle as $\mathcal{A}^{\mathcal{O}}$. CCA security then essentially constitutes preservation of the hiding property even against adversaries enjoying such oracle access. More formally, we define the following game $\mathsf{IND}_b(\langle C, R\rangle, \mathcal{A}, \mathcal{O}, n, z)$ $(b \in \{0,1\})$ as follows: given the public parameter 1^n and auxillary input z, the adversary $\mathcal{A}^{\mathcal{O}}$ adaptively generates two challenge values v_0, v_1 of length n, and a tag $\mathsf{tag} \in \{0,1\}^n$. Then, $\mathcal{A}^{\mathcal{O}}$ receives a commitment to v_b with tag tag from the challenger. Let y be the output of $\mathcal{A}$ in this game. The output of the game is $\perp$ if during the game, $\mathcal{A}$ sends $\mathcal{O}$ any commitment using tag tag. Otherwise, the output of the game is y. We abuse notation to denote the output of the game $\mathsf{IND}_b(\langle C, R\rangle, \mathcal{A}, \mathcal{O}, n, z)$ by the same symbol $\mathsf{IND}_b(\langle C, R\rangle, \mathcal{A}, \mathcal{O}, n, z)$.

▶ **Definition 6** (CCA Commitment). *Let $\langle C, R\rangle$ be a tag-based commitment scheme, and $\mathcal{O}$ be an associated decommitment oracle. Then $\langle C, R\rangle$ is said to be* ***CCA secure w.r.t. $\mathcal{O}$****, if for every nonuniform P.P.T. machine $\mathcal{A}$, the following ensembles are computationally indistinguishable:*

- $\{\mathsf{IND}_0(\langle C, R\rangle, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$
- $\{\mathsf{IND}_1(\langle C, R\rangle, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$

It is customary to call any commitment scheme that is CCA secure with respect to some decommitment oracle as just CCA secure (but in general the oracle is usually also described, and is of course necessary to prove such security). It is also customary to call the interaction between the challenger and adversary as the *left* interaction, and that between adversary and oracle as the *right* interaction, in the fashion of non-malleable commitments, where the security property chiefly considers man in the middle attacks.

Finally, a scheme is *1-1 CCA secure* (denoted as $\text{CCA}^{1:1}$) if the corresponding adversary is only allowed one interaction with the oracle.

3 A New $\text{CCA}^{1:1}$ Commitment Scheme

We will require the following ingredients for our $\text{CCA}^{1:1}$ protocol:

- A statistically-binding commitment Com. Naor's commitment works.
- A 3-round slot-based extractable commitment scheme ExtCom; for concreteness we will use the standard 3-round scheme, shown in Figure 1. based on Naor's commitment (the first message ρ of Naor's commitment is not counted in rounds and assumed to be available from other parts of the protocol).
- An (extractable) commitment scheme ENMC that is *non-malleable* against *synchronizing* adversaries. We will need this protocol to be "compatible with slots" of the ExtCom defined above. For concreteness, we assume that ENMC is the 3-round commitment scheme of [34] which satisfies all our requirements.
- A k round witness indistinguishable argument of knowledge WIAoK.

We stress that all of these ingredients have constant rounds, and can be constructed from standard OWFs in a black-box manner.

Our Protocol. We now describe our first protocol for $\text{CCA}^{1:1}$ commitments. This protocol does not specifically try to achieve the black-box usage of cryptographic primitives. This allows us to focus on proving CCA security. However, it achieves two important properties: it is based on minimal assumptions, and it has a constant number of rounds. Moreover, the structure of this protocol is chosen in such a way that later, it will be possible to convert into a fully black-box construction. We remark that we also directly use identities of length λ directly (this is in keeping with the [34] construction which does the same).

The formal description of the protocol appears in Figure 2. At a high level, the protocol proceeds as follows. First, it requires the receiver to commit to a trapdoor string α using two extractable primitives: ExtCom as well as ENMC. Next, the committer will commit to an all zero-string β using ExtCom. Jumping ahead, in the security proof a "simulator machine" on left will set $\beta = \alpha$ and use it as a "fake witness" in a WIAoK; later we shall instantiate ExtCom with, roughly speaking, a "black-box commit-and-prove" to obtain a black-box construction. The receiver simply opens α in the next step, and the committer commits to the desired value, say v, followed by a proof of knowledge of v or that $\beta = \alpha$. A crucial observation here is that *the proofs are not required to deal with values inside* ENMC – by ensuring that ENMC values opened in the protocol execution.

▶ **Theorem 7.** *The protocol $\langle C, R\rangle_{\mathsf{CCA}}$ (described in Figure 2) is a 1-1 CCA commitment scheme for all polynomial time adversaries.*

The statistical-binding property of protocol $\langle C, R\rangle_{\mathsf{CCA}}$ is straightforward. The computational hiding property is implied by the 1-1 CCA security as per Definition 6. Due to lack of space, we present an outline of the proof for non-malleability below. The complete proof is given in the full version of our paper [16].

Proof for Non-Malleability (Sketch.) We start with a man-in-the-middle adversary $\mathcal{A}$ that participates in the CCA challenge outlined above. Consider any two arbitrary values v_0 and v_1 in the message space. We will now show indistinguishability between the games $\{\mathsf{IND}_0(\langle C, R\rangle_{\mathsf{CCA}}, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$ and $\{\mathsf{IND}_1(\langle C, R\rangle_{\mathsf{CCA}}, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$. To this end, we will use a hybrid argument:

Our proof proceeds as follows. We start with the honest committer on the left committing to some arbitrary value v_0. The overall idea is to move to an intermediate hybrid where the left challenger is able to "set the trapdoor" and go through the WIAoKs without using the commitment. This will allow us to then replace the initial commitment to v_0 to one to v_1 (and move back to doing everything on the left "honestly"). Further, we will also maintain the following invariant across all the hybrids:

▶ **Definition 8** (Invariant Condition (informal))**.** *In the right session, the adversary* MIM *cannot set $\tilde{\beta} = \tilde{\alpha}$ except with negligible probability.*

We outline the necessary hybrids to get to this stage below:

Hybrid H_0^0. This is just the original experiment with the honest committer on the left. In other words, this is the experiment $\{\mathsf{IND}_0(\langle C, R\rangle_{\mathsf{CCA}}, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$. It is straightforward to see that the invariant holds in this hybrid: if it does not, then MIM must break the hiding property of the commitments in Stage 1 or 2 in the right execution to learn $\tilde{\alpha}$.

We let $\lambda \in \mathbb{N}$ denote the security parameter. All primitives used in the protocol by default have 1^λ as part of their input. We omit this detail in the following. Further, we assume that the execution involves a tag or identity $\mathsf{id} \in \{0,1\}^\lambda$.
Input: The committer C and reciever R have common input as the security parameter 1^λ. Additionally, C has as private input a value v which it wishes to commit to.

Commit Phase: This proceeds as follows:

Stage 0: C commits to the value v using Com and sends the identity id to R.

Stage 1: This consists of the following steps:
(a) R picks a value $\alpha \xleftarrow{\$} \{0,1\}^\lambda$.
(b) R commits to $\alpha_1 = \alpha$ using ExtCom.

Stage 2: R commits to $\alpha_2 = \alpha$ using ENMC, using identity id.
For future reference, we denote by $\mathsf{CombinedCom}$ the joint execution of Stage 1 and 2 up to this point. Observe that $\mathsf{CombinedCom}$ is a statistically binding commitment scheme.

Stage 3: C now commits to $\beta = 0^\lambda$ using ExtCom.

Stage 4: This goes as follows:
1. R decommits to both its commitments so far, revealing α_1 and α_2.
2. C checks these decommitments, aborting if $\alpha_1 \neq \alpha_2$.

Stage 5: C and R engage in $k+1$ WIAoK protocols *sequentially*. We denote these WIAoK executions as WIAoK_i for $i = 1, \ldots, k+1$. In all these WIAoKs, C proves the *same* (compound) statement which is true if and only if:
(a) there exists randomness η s.t. $c = \mathsf{Com}(v;\eta)$; **or**
(b) $\beta = \alpha_1 = \alpha_2$, where β is the unique string committed in the transcript of Stage-3.

Note that an honest prover will always use the witness for part-(a) of the above compound statement, which we refer as the "original witness". We will refer the witness for part-(b) of the compound statement. Looking ahead, some hybrids will use the trapdoor witness to go through the WIAoKs.

Decommit Phase: The committer C decommits to v and β. R checks if these decommitments are valid, and accepts if so.

Figure 2 Protocol $\langle C, R\rangle_{\mathsf{CCA}}$: $\mathrm{CCA}^{1:1}$ Commitment Scheme.

Hybrid H_1^0. In this hybrid, the decommitment oracle on the right is removed. All messages are generated as an honest receiver, and the value $\tilde{v}$ committed by the adversary is obtained by extracting the witness from the final WIAoK on the right.

Hybrid H_2^0. This hybrids proceeds as the previous one except that on the left, we also extract the value α_2 from the stage 2 ENMC.

In these two hybrids, the adversary's view up to Stage 4 is identical to that in H_0. Therefore, the invariant must hold in these hybrids (by the same reasoning as in H_0^0). By the knowledge soundness of WIAoK, the extracted witness should be the same as the one returned by the oracle in H_0. The entire view of the adversary is therefore unaffected by these changes, and so is identical to that in H_0^0.

Hybrid H_3^0. This hybrid also proceeds as the previous one except that the value β is now set to be the extracted value α_2.

While it is easy to argue that the adversary cannot detect this change, it may still be able to use the change in the left ExtCom to change its own ExtCom on the right (note that the ExtCom scheme is malleable)! This is the primary reason we need the invariant condition, as it explicitly prevents this exact occurrence. We argue that the invariant holds by considering separate cases for synchronous and nonsynchronous adversaries. For synchronous adversaries, we show roughly that if the adversary could identify any change in the left stage 3 ExtCom, then it must have been influencing the earlier two commitments it made in a malleable fashion - this is ruled out by the non-malleability of ENMC. *As mentioned in the Technical Overview, this is the most difficult case to deal with, and where our main contribution lies.* For nonsynchronous adversaries, we show instead that there are "extraction opportunities" on the left where no messages on the right are exchanged for the corresponding duration (and extraction on the left can be performed unhindered). This relies on carefully setting the appropriate round complexities for ExtCom and ENMC.

Hybrid H_4^0. In this hybrid, we ask the left execution to use the *trapdoor witness* in the Stage 5 WIAoKs (the actual changes happen by constructing a sequence of intermediate hybrids where the witness is replaced one by one in each WIAoK execution on the left, in order of occurrence).

In this hybrid, we ensure that any change of witness on the left does not occur during the same time as extraction of the witness on the right, since the latter involves rewinding and that can interfere with the witness indistinguishability of the left proof. In the synchronizing case, it is easy to see the invariant holds in this hybrid since the corresponding changes all occur *after* the right stage 3 ExtCom is concluded. For nonsynchronous adversaries, this may not be the case, but we can use an argument very similar to that used to argue the invariant in H_3^0. We can also argue indistinguishability of (the view in) this hybrid using the witness indistinguishability of our WIAoK scheme and the fact that the change in witness on the left, and extraction of witness on the right, occur at different times.

Finally, we define H_3^1, H_2^1, H_1^1 to H_0^1, where for H_i^1 is just the analogue of H_i^0, but replacing the initial commitment to v_0 with one to v_1 on the left. Using the same arguments above, we can show that both the invariant condition and indistinguishability views holds among H_4^1, H_3^1 down to H_0^1. Note that H_0^1 is just the experiment $\{\mathsf{IND}_1(\langle C, R\rangle_{\mathsf{CCA}}, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$, and hence we show that this is computationally indistinguishable from $\{\mathsf{IND}_0(\langle C, R\rangle_{\mathsf{CCA}}, \mathcal{A}, \mathcal{O}, n, z)\}_{n\in\mathbb{N}, z\in\{0,1\}^*}$ to $\mathcal{A}$. ◀

4 Our Black-Box CCA Commitment

Our starting point is to determine how we can make our protocol (in Figure 2) fully black-box. In fact, we note that the only component that is not already so is our argument system. Note that we use the arguments to prove statements about the Com in **Stage 0** and the ExtCom in **Stage 3**. Thus, we look to change these components with a suitable "commit-and-prove" protocol that is fully black-box. Further, we will require the following properties from the protocol so that it works with our template:

1. It should provide us a Com, and an ExtCom scheme. Importantly, this "first part" of the protocol should alone serve as a valid commitment with desired properties, regardless of whether we perform a subsequent WIAoK or not. This is in contrast to the definition (and construction) in [48].
2. Later we should be able to perform WIAoK on a compound statement regarding the values committed in the above Com and ExtCom;
3. The WIAoK part should make use of Com and ExtCom only in a black-box manner.

We provide a definition of Commit-and-Prove WIAoK (actually ZKAoK) formally in the full version [16] that captures all the required properties.

We note that there are not many approaches in the literature that achieve what we want: the work of [48] constructed a round-optimal version of such a primitive, but their protocol does not fit our needs because we require some properties that are not immediately avaiable from their construction. We discuss the issues briefly. While their commit stage is a statistically binding commitment, it cannot be modified to be *extractable*, which we crucially need. Further, we will require a *multi-commitment* property for our proof, namely that the predicate to be proved can support values used in *multiple* commit stages; while it is possible that the [48] protocol can be modified to achieve this property, the modification is unclear. Yet another issue is that we will use multiple proof modules (for the same proof) in our final protocol. Again it is not clear how to modify their protocol so that we ensure consistency of openings while also ensuring extractability from every proof module (note that their protocol has a challenge-response format that can allow extraction from just two challenges even across different sessions). We remark however that their protocol does support the *delayed predicate* property, which we also rely on.

We therefore build a commit-and-prove protocol suitable to our purposes. Our starting point is the "MPC-in-the-head" technique from [42]. This approach was originally used to construct black-box zero-knowledge arguments, by having the prover run a virtual MPC execution "in its head" and committing to the views of the virtual parties. The verifier then asks for some of these views to be opened and checks that the opened views are consistent. This construction achieves honest-verifier ZKAoK.[5] To meet our requirements, we want to turn this construction into commit-and-prove form, and bolster the security to tolerate malicious verifiers.

There are a few previous works that already take this approach to create commit and prove protocols. There is the recent work of Hazay, Ishai and Venkitasubramaniam [37], who make use of a commit and prove style protocol constant round secure 2PC protocols against malicious adversaries. Their overall design of the construction utilizes the MPC-in-the-head idea, but by way of using *server watchlists* which is a slightly different implementation of this concept first used in [43]. Being a part of their overall 2PC compiler, their protocol is in the OT-hybrid model, which makes it difficult to adapt to our usage, which is in the plain model. It is also unclear how to modify their construction to have the *multi-commitment* property, as well as make it argument of knowledge, as there is no immediate extraction algorithm to extract the sender's committed value from the proof stage.

We instead follow the approach used in the older work of Goyal et al [31], who use two virtual MPC executions. The first execution is simply a verifiable secret sharing of the value to be committed (the commitment to all the views serves also to commit to this value), which

[5] The authors of [42] also showed how to make it secure against dishonest verifiers. But their technique results in a construction with polynomially-many rounds, because they employ sequential repetition.

is sent on to the verifier (we will call this the *commit phase*). The second execution continues on from the first, where the virtual parties use their shares to compute some predicate on the shared value that represents the statement to be proved over the commitment (we will call this the *proof phase*).

To obtain security against malicious verifiers, one idea is to have the verifier *commit to* its challenge *before* the prover sends its first message. However, we cannot resort to this since this would excise the argument of knowledge property from this construction. We therefore employ a different approach, by building in a *coin-tossing* protocol into the argument system, which only uses an extractable commitment scheme. This is similar to the construction used in [61] to convert ZK arguments to ZKAoKs. It allows the PPT simulator to bias the coin-tossing result, thus allowing for simulation against dishonest verifiers; meanwhile, the knowledge extraction strategy in [42] still works.

The remaining task is to build an extractable commitment scheme that is compatible with the above. To obtain this, we observe that if the predicate ϕ to be simply the identity predicate, then we can still extract the committed value (i.e., the "witness" in the proof phase) as outlined above. Therefore, we view an execution of the above commit-and-prove protocol (with the identity predicate) as an extractable commitment scheme: hiding follows from the hiding of the commit phase as well as zero knowledge of the proof phase, and extractability follows just as mentioned above (i.e., by the argument of knowledge property). We claim further that this commitment scheme is actually *compatible* with our commit-and-prove scheme; this is due to the property of *multiple proofs* mentioned earlier, wherein *separate* proof modules can be performed on the *same* commitments (by adjusting the parameters of verifiable secret sharing). In particular, this allows the prover to use this new extractable commitment scheme, then later perform (black-box) arguments of knowledge for some statement about the value committed to in this scheme. This suffices for our purposes. We provide more details and the formal construction in the full version of our paper [16].

Now we can simply integrate these components into our original 1-1 CCA commitment scheme to obtain a fully black-box instantiation. We present our final protocol and security proof in the full version [16]. Note that our commit-and-prove scheme is constant round, and therefore our final protocol is still constant rounds.

5 Angel-Based MPC in $\widetilde{O}(\log \lambda)$ Rounds

Kiyoshima [50] presents a black-box construction of a CCA-secure commitment scheme with the following ingredients: (a) a two-round statistically-binding commitment scheme, and a constant round "strongly extractable" commitment, both of which are known from (black-box) one-way functions, (b) a concurrently-extractable commitment (due to Micciancio et al., [66]), with a "robustness parameter" ℓ, and (c) an R-round 1-1 CCA-secure commitment provided that $\ell = O(R \cdot \log \lambda \cdot \log\log \lambda)$. The round-complexity of the resulting protocol is $O(\ell)$. If R is a constant, this yields a $\widetilde{O}(\log \lambda)$-round construction for CCA secure commitments.[6] This yields Theorem 2.

As mentioned in the introduction, the security model that we consider is *angel-based* security, or UC *security with superpolynomial helpers*. Very briefly, this is essentially the same as the UC model used in [8], except that the adversary (in the real world) and the

[6] More precisely, Kiyoshima states his results with a specific value of ℓ, namely, $O(\log^2 \lambda \cdot \log\log \lambda)$, since $R = O(\log \lambda)$ in his case. However, his construction and proof work for any value of R if ℓ is as described above.

environment (in the ideal world) both have access to a superpolynomial time functionality that acts as an oracle or a *helper*. Formal definitions for this security model can be found in [12] and [55]. If there is a protocol Π that emulates a functionality $\mathcal{H}$ with helper $\mathcal{H}$ in this setting, we say that Π $\mathcal{H}$-EUC-realizes $\mathcal{F}$.

Now, as in [50], we combine Theorem 2 with the following two results due to Canetti et al. [12, 13] and Lin and Pass [55] respectively to obtain Theorem 1.

▶ **Theorem 9** ([55]). *Assume the existence of an R_{CCA}-round robust CCA-secure commitment scheme $\langle C, R\rangle$ and the existence of an R_{OT}-round semi-honest oblivious transfer protocol $\langle S, R\rangle$. Then, there is an $O(\max(R_{\mathsf{CCA}}, R_{\mathsf{OT}}))$-round protocol that $\mathcal{H}$-EUC-realizes $\mathcal{F}_{\mathsf{OT}}$. Furthermore, this protocol uses $\langle C, R\rangle$ and $\langle S, R\rangle$ only in a black-box way.*

▶ **Theorem 10** ([12, 13]). *For every well-formed functionality $\mathcal{F}$, there exists a constant-round $\mathcal{F}_{\mathsf{OT}}$-hybrid protocol that $\mathcal{H}$-EUC-realizes $\mathcal{F}_{OT}$.*

References

1 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. doi:10.1109/SFCS.2002.1181957.

2 Boaz Barak, Ran Canetti, Jesper Buus Nielsen, and Rafael Pass. Universally composable protocols with relaxed set-up assumptions. In *45th FOCS*, pages 186–195. IEEE Computer Society Press, October 2004. doi:10.1109/FOCS.2004.71.

3 Boaz Barak, Manoj Prabhakaran, and Amit Sahai. Concurrent non-malleable zero knowledge. In *47th FOCS*, pages 345–354. IEEE Computer Society Press, October 2006. doi:10.1109/FOCS.2006.21.

4 Boaz Barak and Amit Sahai. How to play almost any mental game over the net – concurrent composition via super-polynomial simulation. In *46th FOCS*, pages 543–552. IEEE Computer Society Press, October 2005. doi:10.1109/SFCS.2005.43.

5 Donald Beaver. Foundations of secure interactive computing. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 377–391. Springer, Heidelberg, August 1992. doi:10.1007/3-540-46766-1_31.

6 Brandon Broadnax, Nico Döttling, Gunnar Hartung, Jörn Müller-Quade, and Matthias Nagel. Concurrently composable security with shielded super-polynomial simulators. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part I*, volume 10210 of *LNCS*, pages 351–381. Springer, Heidelberg, 2017. doi:10.1007/978-3-319-56620-7_13.

7 Ran Canetti. Security and composition of multiparty cryptographic protocols. *Journal of Cryptology*, 13(1):143–202, January 2000. doi:10.1007/s001459910006.

8 Ran Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd FOCS*, pages 136–145. IEEE Computer Society Press, October 2001. doi:10.1109/SFCS.2001.959888.

9 Ran Canetti and Marc Fischlin. Universally composable commitments. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 19–40. Springer, Heidelberg, August 2001. doi:10.1007/3-540-44647-8_2.

10 Ran Canetti, Oded Goldreich, Shafi Goldwasser, and Silvio Micali. Resettable zero-knowledge (extended abstract). In *32nd ACM STOC*, pages 235–244. ACM Press, May 2000. doi:10.1145/335305.335334.

11 Ran Canetti, Eyal Kushilevitz, and Yehuda Lindell. On the limitations of universally composable two-party computation without set-up assumptions. In Eli Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 68–86. Springer, Heidelberg, May 2003. doi:10.1007/3-540-39200-9_5.

12 Ran Canetti, Huijia Lin, and Rafael Pass. Adaptive hardness and composable security in the plain model from standard assumptions. In *51st FOCS*, pages 541–550. IEEE Computer Society Press, October 2010. doi:10.1109/FOCS.2010.86.

13 Ran Canetti, Huijia Lin, and Rafael Pass. Adaptive hardness and composable security in the plain model from standard assumptions. *SIAM J. Comput.*, 45(5):1793–1834, 2016.

14 Ran Canetti, Yehuda Lindell, Rafail Ostrovsky, and Amit Sahai. Universally composable two-party and multi-party secure computation. In *34th ACM STOC*, pages 494–503. ACM Press, May 2002. doi:10.1145/509907.509980.

15 Nishanth Chandran, Wutichai Chongchitmate, Rafail Ostrovsky, and Ivan Visconti. Universally composable secure computation with corrupted tokens. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part III*, volume 11694 of *LNCS*, pages 432–461. Springer, Heidelberg, August 2019. doi:10.1007/978-3-030-26954-8_14.

16 Rohit Chatterjee, Xiao Liang, and Omkant Pandey. Improved black-box constructions of composable secure computation. Cryptology ePrint Archive, Report 2020/494, 2020. URL: https://eprint.iacr.org/2020/494.

17 Danny Dolev, Cynthia Dwork, and Moni Naor. Non-malleable cryptography (extended abstract). In *23rd ACM STOC*, pages 542–552. ACM Press, May 1991. doi:10.1145/103418.103474.

18 Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *30th ACM STOC*, pages 409–418. ACM Press, May 1998. doi:10.1145/276698.276853.

19 Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *22nd ACM STOC*, pages 416–426. ACM Press, May 1990. doi:10.1145/100216.100272.

20 Sanjam Garg, Vipul Goyal, Abhishek Jain, and Amit Sahai. Bringing people of different beliefs together to do UC. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 311–328. Springer, Heidelberg, March 2011. doi:10.1007/978-3-642-19571-6_19.

21 Sanjam Garg, Vipul Goyal, Abhishek Jain, and Amit Sahai. Concurrently secure computation in constant rounds. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 99–116. Springer, Heidelberg, April 2012. doi:10.1007/978-3-642-29011-4_8.

22 Sanjam Garg, Susumu Kiyoshima, and Omkant Pandey. A new approach to black-box concurrent secure computation. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part II*, volume 10821 of *LNCS*, pages 566–599. Springer, Heidelberg, 2018. doi:10.1007/978-3-319-78375-8_19.

23 Rosario Gennaro, Anna Lysyanskaya, Tal Malkin, Silvio Micali, and Tal Rabin. Algorithmic tamper-proof (ATP) security: Theoretical foundations for security against hardware tampering. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 258–277. Springer, Heidelberg, February 2004. doi:10.1007/978-3-540-24638-1_15.

24 Oded Goldreich, Silvio Micali, and Avi Wigderson. How to play any mental game or A completeness theorem for protocols with honest majority. In Alfred Aho, editor, *19th ACM STOC*, pages 218–229. ACM Press, May 1987. doi:10.1145/28395.28420.

25 Shafi Goldwasser and Leonid A. Levin. Fair computation of general functions in presence of immoral majority. In Alfred J. Menezes and Scott A. Vanstone, editors, *CRYPTO'90*, volume 537 of *LNCS*, pages 77–93. Springer, Heidelberg, August 1991. doi:10.1007/3-540-38424-3_6.

26 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. doi:10.1145/1993636.1993729.

27 Vipul Goyal, Divya Gupta, and Abhishek Jain. What information is leaked under concurrent composition? In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 220–238. Springer, Heidelberg, August 2013. doi:10.1007/978-3-642-40084-1_13.

28 Vipul Goyal, Yuval Ishai, Amit Sahai, Ramarathnam Venkatesan, and Akshay Wadia. Founding cryptography on tamper-proof hardware tokens. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 308–326. Springer, Heidelberg, February 2010. doi:10.1007/978-3-642-11799-2_19.

29 Vipul Goyal and Abhishek Jain. On concurrently secure computation in the multiple ideal query model. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 684–701. Springer, Heidelberg, May 2013. doi:10.1007/978-3-642-38348-9_40.

30 Vipul Goyal, Abhishek Jain, and Rafail Ostrovsky. Password-authenticated session-key generation on the internet in the plain model. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 277–294. Springer, Heidelberg, August 2010. doi:10.1007/978-3-642-14623-7_15.

31 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. doi:10.1109/FOCS.2012.47.

32 Vipul Goyal, Huijia Lin, Omkant Pandey, Rafael Pass, and Amit Sahai. Round-efficient concurrently composable secure computation via a robust extraction lemma. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part I*, volume 9014 of *LNCS*, pages 260–289. Springer, Heidelberg, March 2015. doi:10.1007/978-3-662-46494-6_12.

33 Vipul Goyal, Rafail Ostrovsky, Alessandra Scafuro, and Ivan Visconti. Black-box non-black-box zero knowledge. In David B. Shmoys, editor, *46th ACM STOC*, pages 515–524. ACM Press, 2014. doi:10.1145/2591796.2591879.

34 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. doi:10.1145/2897518.2897657.

35 Jens Groth and Rafail Ostrovsky. Cryptography in the multi-string model. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 323–341. Springer, Heidelberg, August 2007. doi:10.1007/978-3-540-74143-5_18.

36 Iftach Haitner. Semi-honest to malicious oblivious transfer – the black-box way. In Ran Canetti, editor, *TCC 2008*, volume 4948 of *LNCS*, pages 412–426. Springer, Heidelberg, March 2008. doi:10.1007/978-3-540-78524-8_23.

37 Carmit Hazay, Yuval Ishai, and Muthuramakrishnan Venkitasubramaniam. Actively secure garbled circuits with constant communication overhead in the plain model. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part II*, volume 10678 of *LNCS*, pages 3–39. Springer, Heidelberg, November 2017. doi:10.1007/978-3-319-70503-3_1.

38 Carmit Hazay, Antigoni Polychroniadou, and Muthuramakrishnan Venkitasubramaniam. Composable security in the tamper-proof hardware model under minimal complexity. In Martin Hirt and Adam D. Smith, editors, *TCC 2016-B, Part I*, volume 9985 of *LNCS*, pages 367–399. Springer, Heidelberg, 2016. doi:10.1007/978-3-662-53641-4_15.

39 Carmit Hazay and Muthuramakrishnan Venkitasubramaniam. On black-box complexity of universally composable security in the CRS model. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part II*, volume 9453 of *LNCS*, pages 183–209. Springer, Heidelberg, 2015. doi:10.1007/978-3-662-48800-3_8.

40 Carmit Hazay and Muthuramakrishnan Venkitasubramaniam. Composable adaptive secure protocols without setup under polytime assumptions. In Martin Hirt and Adam D. Smith, editors, *TCC 2016-B, Part I*, volume 9985 of *LNCS*, pages 400–432. Springer, Heidelberg, 2016. doi:10.1007/978-3-662-53641-4_16.

41 Yuval Ishai, Eyal Kushilevitz, Yehuda Lindell, and Erez Petrank. Black-box constructions for secure computation. In Jon M. Kleinberg, editor, *38th ACM STOC*, pages 99–108. ACM Press, May 2006. doi:10.1145/1132516.1132531.

42 Yuval Ishai, Eyal Kushilevitz, Rafail Ostrovsky, and Amit Sahai. Zero-knowledge from secure multiparty computation. In David S. Johnson and Uriel Feige, editors, *39th ACM STOC*, pages 21–30. ACM Press, June 2007. doi:10.1145/1250790.1250794.

43 Yuval Ishai, Manoj Prabhakaran, and Amit Sahai. Founding cryptography on oblivious transfer – efficiently. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 572–591. Springer, Heidelberg, August 2008. doi:10.1007/978-3-540-85174-5_32.

44 Abhishek Jain and Omkant Pandey. Non-malleable zero knowledge: Black-box constructions and definitional relationships. In Michel Abdalla and Roberto De Prisco, editors, *SCN 14*, volume 8642 of *LNCS*, pages 435–454. Springer, Heidelberg, September 2014. doi:10.1007/978-3-319-10879-7_25.

45 Yael Tauman Kalai, Yehuda Lindell, and Manoj Prabhakaran. Concurrent general composition of secure protocols in the timing model. In *STOC*, pages 644–653, 2005.

46 Jonathan Katz. Universally composable multi-party computation using tamper-proof hardware. In Moni Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 115–128. Springer, Heidelberg, May 2007. doi:10.1007/978-3-540-72540-4_7.

47 Jonathan Katz and Rafail Ostrovsky. Round-optimal secure two-party computation. In Matthew Franklin, editor, *CRYPTO 2004*, volume 3152 of *LNCS*, pages 335–354. Springer, Heidelberg, August 2004. doi:10.1007/978-3-540-28628-8_21.

48 Dakshita Khurana, Rafail Ostrovsky, and Akshayaram Srinivasan. Round optimal black-box "commit-and-prove". In Amos Beimel and Stefan Dziembowski, editors, *TCC 2018, Part I*, volume 11239 of *LNCS*, pages 286–313. Springer, Heidelberg, November 2018. doi:10.1007/978-3-030-03807-6_11.

49 Joe Kilian. Founding cryptography on oblivious transfer. In *20th ACM STOC*, pages 20–31. ACM Press, May 1988. doi:10.1145/62212.62215.

50 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. doi:10.1007/978-3-662-44381-1_20.

51 Susumu Kiyoshima, Huijia Lin, and Muthuramakrishnan Venkitasubramaniam. A unified approach to constructing black-box UC protocols in trusted setup models. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part I*, volume 10677 of *LNCS*, pages 776–809. Springer, Heidelberg, November 2017. doi:10.1007/978-3-319-70500-2_26.

52 Susumu Kiyoshima, Yoshifumi Manabe, and Tatsuaki Okamoto. Constant-round black-box construction of composable multi-party computation protocol. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 343–367. Springer, Heidelberg, February 2014. doi:10.1007/978-3-642-54242-8_15.

53 Huijia Lin and Rafael Pass. Non-malleability amplification. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 189–198. ACM Press, 2009. doi:10.1145/1536414.1536442.

54 Huijia Lin and Rafael Pass. Concurrent non-malleable zero knowledge with adaptive inputs. In Yuval Ishai, editor, *TCC 2011*, volume 6597 of *LNCS*, pages 274–292. Springer, Heidelberg, March 2011. doi:10.1007/978-3-642-19571-6_17.

55 Huijia Lin and Rafael Pass. Black-box constructions of composable protocols without set-up. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 461–478. Springer, Heidelberg, August 2012. doi:10.1007/978-3-642-32009-5_27.

56 Huijia Lin, Rafael Pass, Wei-Lung Dustin Tseng, and Muthuramakrishnan Venkitasubramaniam. Concurrent non-malleable zero knowledge proofs. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 429–446. Springer, Heidelberg, August 2010. doi:10.1007/978-3-642-14623-7_23.

57 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. doi:10.1007/978-3-540-78524-8_31.

58 Huijia Lin, Rafael Pass, and Muthuramakrishnan Venkitasubramaniam. A unified framework for concurrent security: universal composability from stand-alone non-malleability. In Michael Mitzenmacher, editor, *41st ACM STOC*, pages 179–188. ACM Press, 2009. doi:10.1145/1536414.1536441.

59 Yehuda Lindell. Bounded-concurrent secure two-party computation without setup assumptions. In *35th ACM STOC*, pages 683–692. ACM Press, June 2003. doi:10.1145/780542.780641.

60 Yehuda Lindell. Lower bounds for concurrent self composition. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 203–222. Springer, Heidelberg, February 2004. doi:10.1007/978-3-540-24638-1_12.

61 Yehuda Lindell. A note on constant-round zero-knowledge proofs of knowledge. *Journal of Cryptology*, 26(4):638–654, October 2013. doi:10.1007/s00145-012-9132-7.

62 Tal Malkin, Ryan Moriarty, and Nikolai Yakovenko. Generalized environmental security from number theoretic assumptions. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 343–359. Springer, Heidelberg, March 2006. doi:10.1007/11681878_18.

63 Silvio Micali, Rafael Pass, and Alon Rosen. Input-indistinguishable computation. In *47th FOCS*, pages 367–378. IEEE Computer Society Press, October 2006. doi:10.1109/FOCS.2006.43.

64 Silvio Micali and Leonid Reyzin. Physically observable cryptography (extended abstract). In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 278–296. Springer, Heidelberg, February 2004. doi:10.1007/978-3-540-24638-1_16.

65 Silvio Micali and Phillip Rogaway. Secure computation (abstract). In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 392–404. Springer, Heidelberg, August 1992. doi:10.1007/3-540-46766-1_32.

66 Daniele Micciancio, Shien Jin Ong, Amit Sahai, and Salil P. Vadhan. Concurrent zero knowledge without complexity assumptions. In Shai Halevi and Tal Rabin, editors, *TCC 2006*, volume 3876 of *LNCS*, pages 1–20. Springer, Heidelberg, March 2006. doi:10.1007/11681878_1.

67 Rafail Ostrovsky, Omkant Pandey, and Ivan Visconti. Efficiency preserving transformations for concurrent non-malleable zero knowledge. In Daniele Micciancio, editor, *TCC 2010*, volume 5978 of *LNCS*, pages 535–552. Springer, Heidelberg, February 2010. doi:10.1007/978-3-642-11799-2_32.

68 Rafail Ostrovsky, Silas Richelson, and Alessandra Scafuro. Round-optimal black-box two-party computation. In Rosario Gennaro and Matthew J. B. Robshaw, editors, *CRYPTO 2015, Part II*, volume 9216 of *LNCS*, pages 339–358. Springer, Heidelberg, August 2015. doi:10.1007/978-3-662-48000-7_17.

69 Rafael Pass. Simulation in quasi-polynomial time, and its application to protocol composition. In Eli Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 160–176. Springer, Heidelberg, May 2003. doi:10.1007/3-540-39200-9_10.

70 Rafael Pass. Bounded-concurrent secure multi-party computation with a dishonest majority. In László Babai, editor, *36th ACM STOC*, pages 232–241. ACM Press, June 2004. doi:10.1145/1007352.1007393.

71 Rafael Pass, Huijia Lin, and Muthuramakrishnan Venkitasubramaniam. A unified framework for UC from only OT. In Xiaoyun Wang and Kazue Sako, editors, *ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 699–717. Springer, Heidelberg, December 2012. doi:10.1007/978-3-642-34961-4_42.

72 Rafael Pass and Hoeteck Wee. Black-box constructions of two-party protocols from one-way functions. In Omer Reingold, editor, *TCC 2009*, volume 5444 of *LNCS*, pages 403–418. Springer, Heidelberg, March 2009. doi:10.1007/978-3-642-00457-5_24.

73 Birgit Pfitzmann and Michael Waidner. A model for asynchronous reactive systems and its application to secure message transmission. In *2001 IEEE Symposium on Security and Privacy*, pages 184–200. IEEE Computer Society Press, May 2001. doi:10.1109/SECPRI.2001.924298.

74 Manoj Prabhakaran, Alon Rosen, and Amit Sahai. Concurrent zero knowledge with logarithmic round-complexity. In *43rd FOCS*, pages 366–375. IEEE Computer Society Press, November 2002. doi:10.1109/SFCS.2002.1181961.

75 Manoj Prabhakaran and Amit Sahai. New notions of security: Achieving universal composability without trusted setup. In László Babai, editor, *36th ACM STOC*, pages 242–251. ACM Press, June 2004. doi:10.1145/1007352.1007394.

76 Ransom Richardson and Joe Kilian. On the concurrent composition of zero-knowledge proofs. In Jacques Stern, editor, *EUROCRYPT'99*, volume 1592 of *LNCS*, pages 415–431. Springer, Heidelberg, May 1999. doi:10.1007/3-540-48910-X_29.

77 Alon Rosen. A note on constant-round zero-knowledge proofs for NP. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 191–202. Springer, Heidelberg, February 2004. doi:10.1007/978-3-540-24638-1_11.

78 Hoeteck Wee. Black-box, round-efficient secure computation via non-malleability amplification. In *51st FOCS*, pages 531–540. IEEE Computer Society Press, October 2010. doi:10.1109/FOCS.2010.87.

79 Andrew Chi-Chih Yao. How to generate and exchange secrets (extended abstract). In *27th FOCS*, pages 162–167. IEEE Computer Society Press, October 1986. doi:10.1109/SFCS.1986.25.