Electronic Colloquium on Computational Complexity, Report No. 164 (2012)

Simultaneous Resettability from Collision Resistance

Rafail Ostrovsky
UCLA
USA

Ivan Visconti
University of Salerno
ITALY

Abstract

In FOCS 2001, Barak, Goldreich, Goldwasser and Lindell conjectured that the existence of ZAPs, introduced by Dwork and Naor in FOCS 2000, could lead to the design of a zero-knowledge proof system that is secure against both resetting provers and resetting verifiers. Their conjecture has been proven true by Deng, Goyal and Sahai in FOCS 2009 where both ZAPs and collision-resistant hash functions (CRHFs, for short) play a fundamental role.

In this paper, we present a new technique that allows us to prove that simultaneously resettable zero knowledge can be achieved by relying on CRHFs only. Our construction therefore goes beyond the conjecture of Barak et al. bypassing the (demanding) use of ZAPs, that in turn require double enhanced trapdoor permutations (DTPs, for short). More specifically, we present the following results:

1. We construct the first resettably-sound resettable witness indistinguishable (rsrWI, for short) argument for $\mathcal{NP}$ based on CRHFs. Our construction exploits a new technique that we call "soundness upgrade". In order to upgrade stand-alone soundness to resettable soundness, we use the lower bound proved by Rosen in CRYPTO 2000 on the round complexity of black-box concurrent zero knowledge. Moreover our rsrWI argument is an argument of knowledge (AoK, for short).
2. As an application of the above result, we obtain the main theorem of this work: we prove (constructively) the existence of an argument system that is both resettable zero knowledge and resettably sound under the sole assumption that CRHFs exist.

Our results improve the state-of-the-art, and, perhaps even more importantly, provide a novel tool for the design of resettably-secure protocols. We also show a novel way to use protocol lower bounds in constructive protocol design.

Keywords: Proof Systems, Resettable WI/ZK/Soundness.

1 Introduction

Reset attacks, in the context of interactive proofs [GMR85], were introduced by Canetti, Goldreich, Goldwasser, and Micali [CGGM00]. Launching a reset attack on a machine B means that the adversary A can interact with B many times, each time resetting B to its initial state and forcing B to use the *same* random coins. Essentially, this is equivalent to A having access to many *identical* copies of B, i.e., all with same initial configuration and random tape; A can interact with these copies in an arbitrary manner.

These attacks are often easy to launch, particularly when the devices do not have an undetachable power supply such as a smart card or an RFID chip. Due to both theoretical and practical importance, reset attacks have received considerable attention.

Reset attacks are difficult to protect against. As shown in [CGGM00], with appropriate modeling, security against reset attacks implies security against "concurrent" attacks [DNS98]. Many important cryptographic protocols have been designed that retain some form of "security" against reset attacks, for example: zero-knowledge interactive proofs [CGGM00, BGGL01, MR01a, MR01b, BLV03, ZDLZ03, CPV04a, CPV04b, APV05, BLV06, YZ07, DL07, VV09, DGS09, DFG+11, BOV12, COSV12, GOVW12, SV12, BP12], identification schemes [BFGM01,

ISSN 1433-8092

BPSV08, COSV12], secure computation [GS08, GM11, GKOV12], and public-key encryption [Yil10].

In this paper, we continue the study of reset attacks in proof systems, and in particular we focus on the simultaneous resettability conjecture of Barak, Goldreich, Goldwasser and Lindell [BGGL01], that we will refer to as the *BGGL conjecture*:

"Do languages outside of BPP have resettably-sound arguments that are resettable zero-knowledge? Some hope for an affirmative resolution of the above question is provided by the fact that some level of resettable security for both parties does seem to be achievable" (i.e., Zaps).

The BGGL conjecture stresses that the existence of ZAPs, introduced by Dwork and Naor [DN00, DN07], that are rsrWI proofs gives hope to finally obtain a resettably-sound resettable zero-knowledge (rsrZK, for short) argument. Without ZAPs, it was known how to protect one-side only from reset attacks, namely: prover's protection only through resettable zero knowledge arguments (rZK, for short) and verifier's protection only through resettably-sound zero-knowledge arguments (rsZK, for short).

The BGGL conjecture has been shown to be true in the recent result of Deng, Goyal and Sahai, [DGS09], where simultaneous resettability is achieved for $\mathcal{NP}$ under the assumption that CRHFs and ZAPs [DN00] exist. The former assumption is due to the use of non-black-box techniques, extending the previous work of Barak [Bar01] under standard security (here we ignore the controversial non-standard "knowledge of exponent" assumption and other variations that deviate from the standard notion of computational-complexity based security).

One of the key building blocks in [DGS09] consists in proving that a given statement is true by means of a proof system that is both resettably sound and resettable witness indistinguishable. ZAPs [DN00] are the only proof systems that satisfy such a security guarantee with respect to reset attacks. ZAPs can be constructed under the existence of NIZK proofs for $\mathcal{NP}$ [FLS90], therefore by relying on DTPs (see [Gol11]). In contrast, rZK arguments [CGGM00] and rsZK arguments can be constructed for all $\mathcal{NP}$ by assuming the sole existence of CRHFs.

A few weeks ago, an alternative construction for one-sided resettability, namely rsZK arguments based on oblivious transfer has been presented in [BP12] (recall that oblivious transfer can be based either on enhanced trapdoor permutations or homomorphic encryption, in contrast to CRHFs that do not require a trapdoor). In contrast, our paper aims at simultaneous resettability.

1.1 Difficulties and the Power of ZAPs

Designing a zero-knowledge protocol that is secure against reset attacks requires to address the following issues: 1) The simulator must have an advantage over a malicious resetting prover, therefore the simulator must be non-black-box. 2) There must be a proof of soundness that does not just rely (e.g., it could rely on information-theoretic techniques) on rewinding the malicious prover as a black box. Indeed the same could be exploited by the adversarial verifier that can reset the honest prover in order to gain information to break resettable witness indistinguishability (rWI, for short) or rZK.

The approach of Canetti et al. for rZK. The construction of [CGGM00] overcomes the latter issue by immunizing the prover from resets of the adversarial verifier and relying on unconditional soundness (i.e., to prove soundness there is no reduction that uses the malicious prover as a black box). More specifically, the paradigm proposed by [CGGM00] to design rZK proofs, consists in requiring the verifier to commit to all its next messages when the protocol starts. This special message is referred to as the "determining message". Later on the verifier will just open round by round what he committed in the determining message. Therefore the malicious verifier V^* does not obtain any advantage in rewinding the prover unless the determining message is changed. The case in which V^* resets the prover and changes the determining message is not

an issue. Indeed, [CGGM00] shows that the prover can simply use a pseudo-random function (PRF, for short) on input the determining message, in order to determine the randomness to be used later. This basically means that such a reset of V^* that corresponds to a new determining message produces a computationally independent new session. This protects the zero-knowledge property against reset attacks. However even though soundness can still be proved by using information-theoretic arguments (i.e., unconditional soundness) there is no protection against a resetting prover P^*. The problem is that the honest verifier is stuck with the determining message and thus a resetting prover can take advantage of it. For instance P^* could just run the very same code of the simulator that in [CGGM00] in a black-box fashion simply rewinds the verifier. Thus protocol of [CGGM00] is clearly not resettably sound.

The approach of [BGGL01] for rsWI/rsZK. The construction of rsWI/rsZK given in [BGGL01] takes the opposite direction, and relies on the fact that in contrast to the construction of [CGGM00], all messages of the verifier are computationally unrelated to each other. This makes useless the resetting capabilities of P^* and can be implemented by asking the verifier to use a PRF on input the transcript so far. More specifically, [BGGL01] shows a transformation that on input a constant-round public-coin witness-indistinguishable (WI, for short)/zero-knowledge (ZK, for short) argument (e.g., Blum's protocol [Blu86]/Barak's protocol [Bar01]) outputs a new protocol. In the new protocol the prover is untouched and the verifier uses a PRF on input the transcript so far instead of public coins. Informally, resettable soundness follows from the fact that messages played before the reset are (computationally) independent (and thus useless for P^*) of messages played after the reset[1]. Instead, the WI/ZK of the original protocol is preserved in the transformed protocol because whatever the new adversarial verifier can do on the transformed protocol, could also be done on the original protocol. At the same time their construction gives a lot of freedom to a resetting verifier since it can reset the prover and change any message in the transcript. This can be typically used to reset the prover and break the WI/ZK property, for instance extracting a witness. The protocol of [BGGL01] is clearly not rWI/rZK.

Interaction is Achilles' heel w.r.t. simultaneous resets. As just explained, both [BGGL01] and [CGGM00] are easy to attack through simultaneous resets. The reason is that the main ideas of both papers work on one side only, and make those protocols completely vulnerable with respect to attacks from the other side. This seems to be an evident limitation of interactive protocols which makes (at a first glance) the use of ZAPs unavoidable. That is, it appears that through careful protocol design (i.e., a verifier that commits from the beginning, or a completely uncommitted verifier) and relying on special forms of zero knowledge (i.e., non-black-box zero knowledge) and special forms of soundness (i.e., unconditional soundness) it seems that protection from reset attacks is achievable on one side only, unless ZAPs are deployed.

Thus, given the evident difficulties in achieving security under simultaneous resettability, the BGGL conjecture heavily relied on the fact that ZAPs are already secure (but in the WI sense only) against resets on both sides. The reason why the above difficulties do not apply to ZAPs is that ZAPs are essentially non-interactive, and thus resets against ZAP players are useless. Therefore by relying on the existence of such extremely powerful proof systems one might be able to make sure that both techniques of [CGGM00] and of [BGGL01] can co-exist in one protocol only, protecting the weaknesses of [CGGM00] and [BGGL01] through some use of ZAPs. This was precisely exploited in the important result of [DGS09], where the technique of [CGGM00] is combined in a non-trivial way with the one of [BGGL01] by critically using ZAPs, previous rewinding techniques and a new non-black-box technique.

[1] This is actually true only combined with the constant-round requirement that forces the existence of at least one long message played by the verifier and that is critical for soundness.

In [DGS09] there is a sort of determining message for the prover too. Indeed, the prover commits to the randomness to be used in its next messages. In order to make simulation possible, the [DGS09] protocol allows a prover/simulator to deviate from the committed messages (and thus to exploit rewinds) only when the statement is true (i.e., "$x \in L$") or a certain trapdoor theorem is true. The simulator manages to make that trapdoor theorem true by using knowledge of the code of the adversarial verifier. Notice that this approach gives resettable soundness since P^* wants to prove a false statement and does not know the code of the verifier. Therefore during the attack of P^* both conditions are not satisfied and thus resets are useless. The simulator instead will make use of the code of the verifier to make true the trapdoor theorem, and then can also take advantage of rewindings, since it can (in contrast to P^*) deviate from the committed randomness. However the strategy of [DGS09] does not work by just applying the above idea to [CGGM00] and [BGGL01]. Indeed, the above technique needs a subprotocol that allows to prove that at least one among some theorems (e.g., "the played message is consistent with the committed randomness", "$x \in L$", "the trapdoor theorem is true") is true, in a setting where both parties (if corrupted) can have reset capabilities. Again having an interactive protocol for this task is considered an important open problem. Fortunately [DGS09] can rely on the non-interactive power of ZAPs that are therefore proof systems resilient to reset attacks in both directions and remain witness indistinguishable (WI, for short) only, therefore rsrWI. WI is precisely what was needed in [DGS09] in order to make indistinguishable the case in which the code of the adversary is exploited by the simulator that makes the trapdoor theorem true, from the case where a honest prover plays in the experiment that therefore includes a false trapdoor theorem.

The cost of ZAPs and the open problem. As already explained, the whole approach of [DGS09] relies on ZAPs, that being (essentially) non-interactive WI proof systems that are resilient to reset attacks. However the same fact that they are (essentially) non-interactive makes them hard to construct. Indeed, so far we can construct ZAPs from NIZK by only relying on very strong complexity-theoretic assumptions with trapdoors, such as DTPs.

From the above discussion, an important open question is therefore the existence of WI/ZK arguments resilient to simultaneous reset attacks without relying on ZAPs, therefore implementable under less demanding computational assumptions. In turn, another central open problem is the study of the minimal complexity assumptions to achieve protocols that are secure with respect to reset attack in both directions.

1.2 Our Contribution

In this paper we shed light on the above open question by proving that under the sole assumption of the existence of CRHFs, one can construct a rsrZK argument for $\mathcal{NP}$. Surprisingly, this goes beyond the BGGL conjecture and we achieve this by introducing a new technique to defeat reset attacks in *interactive* protocols. In addition, our method allows us to remove ZAPs from the previous work of [DGS09].

Obtaining the above main result goes through a sequence of three steps that are also of interest (and improve previous works) in their own right, as we explain below.

Soundness upgrade. The first step that we achieve is the main technical contribution of this work: we show a transformation from stand-alone soundness to resettable soundness in a more general setting than the one of [BGGL01] as we describe below. In [BGGL01] it is showed how to convert any constant-round *public-coin* argument system into a resettably-sound argument system preserving the round complexity, the WI/ZK and AoK properties, under the assumptions that one-way functions exist. The existence of a BGGL-like transformation for *private-coin* protocols based on CRHFs only is an open problem. We remove the "public-coin" producing

a new transformation that as input only needs a WI/rWI protocol with at most 7 rounds[2]. Our transformed protocol will be resettably-sound, however will have polynomial round complexity. The technique that we use for this result is new and could have other applications. Indeed, in our transformation the honest verifier will run the adversary of concurrent zero knowledge shown in [Ros00] where Rosen proved that black-box concurrent zero knowledge can not be achieved in 7 rounds. The new transformation can be applied to (any) 7-round AoK in order to obtain a resettably-sound AoK, preserving WI/rWI.

Our technique shows how to use a lower bound on the round complexity of black-box concurrent zero-knowledge to actually design another WI/rWI protocol with very strong security guarantees.

rsrWI for $\mathcal{NP}$ from CRHFs. The second step consists of designing a 7-round rWIAoK from CRHFs. To obtain this result we carefully use Blum's protocol [Blu86] and Barak's protocol [Bar01] along with the Lapidot and Shamir protocol [LS90]. The resulting construction is a 7-round rWIAoK and can therefore be used as input to our new transformation, therefore obtaining the first rsrWIAoK based on CRHFs only.

rsrZK arguments from CRHFs. The third step produces the main claim of this paper. We show a rsrZKAoK under the sole assumption that CRHFs exist, thus improving the recent work of [DGS09]. To obtain this result we start from the protocol of [DGS09] and replace the ZAP with the rsrWIAoK based on CRHFs. We can not obtain the desired claim directly since there is another important issue to address. ZAPs are proof systems, while what we obtained using CRHFs is an AoK. Therefore after replacing ZAPs in the protocol of [DGS09] with a rsrWI argument of knowledge we must also add proper reductions to the security proof of [DGS09] in order to state the main claim.

We stress that all our constructions do not impose any a priori bound on the number of resets and we follow the standard modeling also used in [DGS09].

2 Definitions and Tools

The definitions of rZK and rWI can be found in [CGGM00] and in Appendix A.3. The definitions of rsZK, rsWI, rsAoK, can be found in [BGGL01] and in Appendix A.4. The definitions of rsrWI (resp., rsrZK) simply consists in requiring both rWI (resp., rZK) and rsWI (resp., rsZK) hold for the same proof system. The definition of a universal argument can be found in [BG02] and in Appendix A.5. The definition of a ZAP can be found in [DN00].

3 Technical Overview

We now give a technical overview of our techniques and results explaining in details all main ideas. In Section 4, Section 5 and Section 6 we prove formally all our claims.

We now proceed backward compared to the description given in the Section 1.

From ZAPs to rsrWIAoK. We start by considering the rsrZK construction of [DGS09]. We notice that their construction critically relies on the use of ZAPs (and thus of NIZK, which in turn involves the use of DTPs). A more careful analysis of [DGS09] however reveals that their

[2]In fact it is plausible that our technique can be extended to a super-constant number of rounds, still below $\tilde{\Omega}(\log n)$ rounds. We did not further investigate this possibility as 7 rounds are sufficient for our application, i.e., rsrZK from CRHFs.

construction still works by replacing ZAPs with any rsrWI proof systems[3]. Additionally, we observe that the security proof of [DGS09] can be updated by adding some reductions so that the rsrWI proof can be replaced by any rsrWIAoK. The details of the above discussion can be found in Section 6 where we show how to replace in [DGS09] the use of ZAPs by any rsrWIAoK still obtaining a rsrZK argument systems (actually we do even better since we obtain a rsrZKAoK). In light of this, the main technical problem, consists in constructing a rsrWIAoK from CRHFs only.

In order to construct a rsrWIAoK from CRHFs, we notice that resettable soundness was achieved in [BGGL01] by transforming any constant-round public-coin ZK/WI argument into a rsZK/rsWI argument with the same round complexity. However, to obtain a rsrWIAoK, we would need to apply the [BGGL01] transformation to a rWIAoK. Unfortunately, regardless of the round complexity, we do not know how to construct a public-coin rWIAoK from CRHFs (the only known constructions are ZAPs and require DTPs), and therefore we can not use the [BGGL01] transformation to obtain a rsrWIAoK relying on CRHFs only.

The above discussion implies that we need a completely new technique that indeed will be the main technical contribution of this work. We will propose a new transformation that adds resettable soundness to any (i.e., even private coins) 7-round rWIAoK.

Therefore there are two results that we describe next: a) our new transformation, and b) a 7-round rWIAoK.

7-round rWIAoK. We construct a 7-round rWIAoK by considering the following steps. Take Blum's protocol [Blu86] for Hamiltonicity. It consists of the following 3 rounds: 1) commitments of permuted adjacency matrixes, 2) random challenge, and 3) opening of some of the committed bits in the matrixes according to the challenge.

The first update that we make consists in applying the CGGM transformation [CGGM00] to the above protocol, asking the verifier to commit to the challenge first, with a statistically hiding commitment and asking the prover to use a PRF on input the above commitment. This produces a 5 round rWI proof system, where in the 4-th round the verifier opens the committed challenge, therefore sending both the challenge and the associated decommitment information.

The second update, following [BGGL01], consists in replacing the above decommitment information with a resettably-sound statistical ZK argument of knowledge using CRHFs [PR05a]. In this AoK the verifier proves knowledge of the associated decommitment information. The resulting protocol is a constant-round rWIAoK.

While the above protocol is constant round, the actual number of rounds is higher than 7. We then parallelize subprotocols and make use of the Lapidot and Shamir [LS90] Hamiltonicity proof system to reorganize all involved messages into a 7-round protocol. Along the way, this will produce a 6-round public-coin statistical ZKAoK, and a 6-round statistical rsZKAoK.

Soundness upgrade. Let $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ be the above 7-round protocol. We use in the novel (and constructive) way the lower bound for 7-round concurrent zero knowledge of [Ros00] to produce a new protocol that is also resettably sound. Recall that in [Ros00], the lower bound is proved by showing an adversarial verifier $V^*_{\mathtt{rWI}}$ that opens a polynomial number of sessions using a specific scheduling. It is proved in [Ros00] that (assuming the language is non-trivial) a black-box simulator can succeed in generating an indistinguishable transcript only if at some point it manages to prove one of the sessions (either in the main thread of the simulation where the final transcript is constructed, or during some sessions played in other threads due to rewindings which messages will not appear in the final transcript) in straight line.

[3]Because of difficulties to protect an interactive protocol from reset attacks in both directions, the only rsrWI proof that we know is (essentially) non-interactive, i.e., ZAPs.

Our transformation produces a protocol $\pi = (P_\pi, V_\pi)$ where the honest verifier V_π plays precisely as adversarial strategy of concurrent verifier $V^*_{\mathtt{rWI}}$ in [Ros00], therefore a protocol session will include several sub-sessions scheduled according to the strategy of $V^*_{\mathtt{rWI}}$, and the prover $P_{\mathtt{rWI}}$ has to convince $V_{\mathtt{rWI}}$ in all sub-sessions in order to convince V_π.

There is a technical issue to discuss here. $V^*_{\mathtt{rWI}}$ in [Ros00] uses a t-wise independent hash function in order to guarantee that the randomness used after a rewind is independent of the randomness used before the rewind (unless the prefix of the session is identical). Since the proof of [Ros00] aims at showing that for any black-box simulator there exists a $V^*_{\mathtt{rWI}}$ such that the simulator is forced to complete at least one sub-session in straight line, the value t of the t-wise independent hash function can be based on the running time of the simulator. This choice guarantees that the simulator can not get advantage from relations among re-started sub-sessions. Indeed $V^*_{\mathtt{rWI}}$ by using sufficient randomness and the t-wise independent hash function will play the re-started sub-sessions as perfectly independent sub-sessions. Unfortunately our transformation can not follow this convenient tool used by [Ros00] since in our case we must show a protocol that works against any resetting prover. Therefore we first have to fix $V_\pi = V^*_{\mathtt{rWI}}$ which in turn implies that we have first to fix some value t. But once t is fixed, there always exists a malicious prover P^*_π that starts more than t sub-sessions and manages to predict the randomness of $V^*_{\mathtt{rWI}}$ inside V_π, therefore violating the resettable soundness of π. We will therefore ask V_π to run $V^*_{\mathtt{rWI}}$ but using a PRF instead of a t-wise independent hash function. This of course will require an additional delicate argument in the proof of resettable soundness that reduces a successful P^*_π to a forgery for the PRF. In the following discussion when referring to $V^*_{\mathtt{rWI}}$ we will actually refer to the adversarial verifier of [Ros00] but replacing the t-wise independent hash function with a PRF.

Now, notice that when by contradiction P^*_π is successful for a false theorem "$x \in L$", it succeeds to prove that false theorem in a session of π, which means in all played sub-sessions. In particular such sub-sessions include the completed sub-sessions played in the main thread (appearing in the final transcript that makes P^*_π successful) and other threads generated by resets (that are then discarded by P^*_π when producing a successful transcript).

As discussed before regarding the black-box simulator, the analysis of [Ros00] already proves that if P^*_π can do so, then it will at some point solve a sub-session in straight-line. This sub-session will play a crucial role in the proof of resettable soundness. Indeed, we will run the extractor of the rWIAoK $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ on this sub-session. The reason this is critical is that if we run the extractor in a different sub-session, then P^*_π by resetting V_π would reset also the extractor which means that P^*_π can actually distinguish a run with the extractor inside V_π with respect to a run with $V^*_{\mathtt{rWI}}$ only inside V_π. Then P^*_π would abort the experiment. In our proof, since the extraction will be played in that special (i.e., straight-line) sub-session, P^*_π will not deviate its behavior and extraction will be possible. Notice moreover that the extraction procedure will require to use pure randomness instead of the previously mentioned PRF used by honest verifier V_π. This will be addressed in our hybrid arguments.

It is easy to see that the transformation preserves rWI since rWI of $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ already assumes that $V^*_{\mathtt{rWI}}$ can ask for the execution of many sessions for any scheduling of its choice, and the transformation to π is therefore safe with respect to such attacks. [4]

[4] We finally notice that our approach might also work without the restriction of $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ to be a 7-round protocol, as long as the round complexity does not reach the lower bound for black-box concurrent ZK proved in [CKPR01]. This would require to address some additional issues. Indeed in this case the adversarial verifier of [CKPR01] $V^*_{\mathtt{rWI}}$ would not complete all sub-sessions and thus V_π should not expect all sub-sessions completed. The reason is that the proof of [CKPR01] in contrast to the one of [Ros00] critically relies on an aborting adversarial verifier $V^*_{\mathtt{rWI}}$. When included in our transformation, this verifier might give an additional advantage to P^*_π that can take advantage of some transcripts with many aborts. We did not further investigate on this, since the transformation based on the lower bound of [Ros00] is sufficient to prove our main theorem.

4 Soundness Upgrade

In Fig. 1 we show our rsrWIAoK $\pi = (P_\pi, V_\pi)$. It uses k executions of a 7-round rWIAoK $\pi_{\mathtt{rWI}} = (P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$. We denote the i-th message of $P_{\mathtt{rWI}}$ (resp., $V_{\mathtt{rWI}}$) in a sub-session of π by $P_{\mathtt{rWI}i}$ (resp., $V_{\mathtt{rWI}i}$). The picture shows 8 messages for such a subprotocol but the 8-th round is just the final output of the verifier that is shown as a message for convenience only. The scheduling of the k sub-sessions is established recursively according to a variable m which value changes at each recursion. The initial value of m is k. The randomness used by V_π in each of the k sub-sessions is obtained by using a PRF on input the sequence of messages sent by the prover in the execution of π until the first round of the sub-session. V_π aborts if in any moment there is an invalid message from P_π or if there is a $V_{\mathtt{rWI}4}$ ="reject".

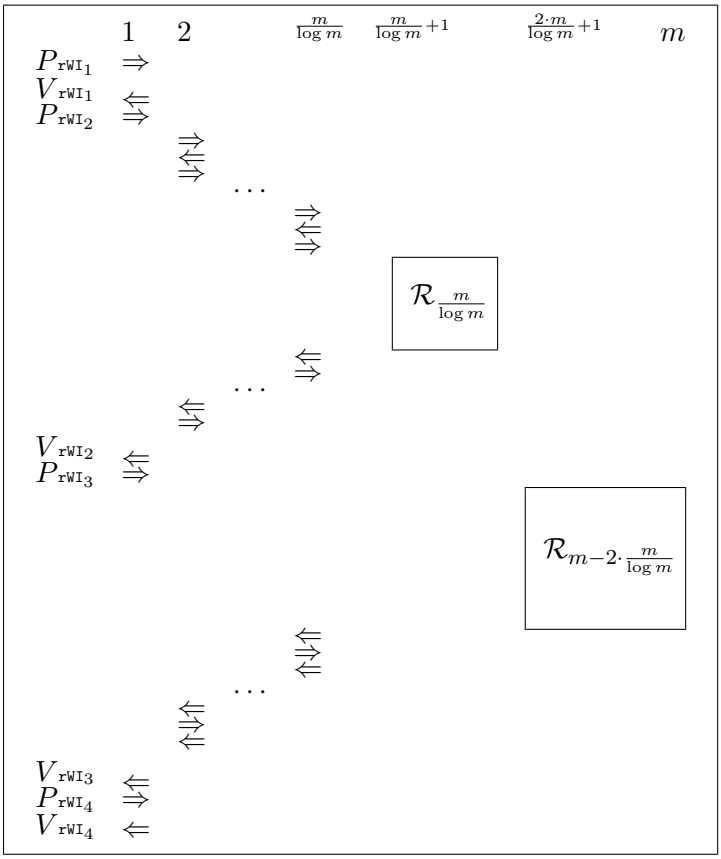

Figure 1: The execution of protocol $\pi_{\mathtt{rWI}}$ between a prover $P_{\mathtt{rWI}}$ and a verifier $V_{\mathtt{rWI}}$.

Theorem 1 *The argument system (P_π, V_π) described above is a resettably-sound resettable witness indistinguishable argument of knowledge.*

Proof. Towards proving resettable soundness of π, we start making use of Lemma 11 of [Ros00]. There, it is proved that according to the scheduling of $V^*_{\mathtt{rWI}}$, if a simulator has always to rewind a sub-session to have a non-negligible probability of completing it successfully, then the running time of the simulator is superpolynomial. This holds because according to the scheduling, the total work that the simulator would do is described by the following recursion:

$$W(m) \geq \mathtt{min}\{\frac{m}{\log m} W(\frac{m}{\log m}), 2W(m - 2\frac{m}{\log m})\}$$

and therefore $W(k) = k^{\Omega(\frac{\log k}{\log\log k})}$.

Notice that we can not apply this analysis to P^*_π since the protocol π does not perfectly correspond to the execution of k sub-sessions of $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ with the scheduling established by $V^*_{\mathtt{rWI}}$. Indeed, in π we have introduced a PRF to replace the use of the t-wise independent hash function, and thus V_π does not behave as $V^*_{\mathtt{rWI}}$. We now apply the analysis to a different verifier V^t_π, and then we will consider the case of V_π.

Let t be the max number of steps of P^*_π before stopping. Consider a PPT machine V^t_π that activates P^*_π and answers to its queries by emulating the honest verifier V_π, with the following exception: each time V_π queries the PRF with a new input, the actual output that will be used by V^t_π is the output of a t-wise independent hash function mapping $\mathtt{poly(k)}$-bit long sequences to the number of random bits needed in one execution of $V_{\mathtt{rWI}}$.

The above behavior of V^t_π is by inspection identical to the one of the successful adversarial verifier $V^*_{\mathtt{rWI}}$ of $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$.

We define the session-prefix of a sub-session $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ of π as the entire transcript of messages in the view of V_π until the first round of that sub-session. Notice that a reset can force the verifier to play again the same sub-session with a different session-prefix. The use of the t-wise independent hash function makes completely independent these two executions of the same sub-session.

We can now claim that P^*_π can convince V^t_π on an instance x with non-negligible probability *only if* during the execution of the experiment started by V^t_π there is at least one sub-session for a given session-prefix that has been completed successfully in straight line.

Lemma 1 *Assume that V^t_π accepts for some x the proof given by P^*_π with non-negligible probability. Then there is at least one sub-session with a session-prefix q that has been successfully completed on common input x with a transcript $(P_{\mathtt{rWI}1},V_{\mathtt{rWI}1},\ldots,P_{\mathtt{rWI}4},V_{\mathtt{rWI}4})$ and in the whole experiment V^t_π played in that sub-session with session-prefix q those 4 messages only.*

Proof. Assume by contradiction that the claim does not hold. Therefore we have that for every different session-prefix of each sub-session, there always is an index $i \in \{1,2,3,4\}$ such that P^*_π received two messages $V_{\mathtt{rWI}i}$. By construction, we know that in order to get the second $V_{\mathtt{rWI}i}$, P^*_π has to complete again all sub-sessions in between $P_{\mathtt{rWI}i}$ and $V_{\mathtt{rWI}i}$, and they are all played with a different session-prefix.

Therefore as explained above, following the analysis of [Ros00], we have that the running time of P^*_π is at least $k^{\Omega(\frac{\log k}{\log\log k})}$, which is a contradiction since P^*_π is PPT.

Consider now a PPT machine $V^{\mathtt{rand}}_\pi$ that emulates the honest verifier V_π to P^*_π in the experiment of resettable soundness, with the following exception: each time V_π queries the PRF with a new input, then the actual output that will be used by $V^{\mathtt{rand}}_\pi$ is purely random.

Lemma 2 *Assume that $V^{\mathtt{rand}}_\pi$ accepts for some x the proof given P^*_π with non-negligible probability. Then there is at least one sub-session with a session-prefix q that has been successfully completed on common input x with a transcript $(P_{\mathtt{rWI}1},V_{\mathtt{rWI}1},\ldots,P_{\mathtt{rWI}4},V_{\mathtt{rWI}4})$ and in the whole experiment $V^{\mathtt{rand}}_\pi$ played in that sub-session with session-prefix q those 4 messages only.*

Proof. Notice that V^t_π uses a t-wise independent hash function and the function is queried less than t times. Therefore it holds that V^t_π and $V^{\mathtt{rand}}_\pi$ coincide since they consist in replacing in V_π the output of each new invocation of the PRF with a new random string. Since by Lemma 1 the claim holds for V^t_π, we have that it holds for $V^{\mathtt{rand}}_\pi$ too.

We say that an extractor is *canonical* if it first runs as a honest verifier to obtain the statements proven successfully, and then it outputs witnesses for those statements. We now prove that there exists a *canonical* extractor $E^{\mathtt{rand}}_\pi$ for the instances proved by P^*_π to $V^{\mathtt{rand}}_\pi$.

Lemma 3 *There exists a* canonical *extractor $E^{\mathtt{rand}}_\pi$ that except with negligible probability, outputs pairs (x,w) so that:*

1. *x is distributed identically to the instances proved in the above execution of P_π^* with $V_\pi^{\mathtt{rand}}$;*
2. *w is a legal $\mathcal{NP}$ witness for x.*

Proof. $E_\pi^{\mathtt{rand}}$ simply runs the code of $V_\pi^{\mathtt{rand}}$ with P_π^*. During interaction, for every instance x successfully proved by P_π^*, $E_\pi^{\mathtt{rand}}$ stores the session-prefixes of the straight-line sub-sessions. Next, for each such session-prefix, sequentially, $E_\pi^{\mathtt{rand}}$ runs the extractor $E_{\mathtt{rWI}}$ of the rWIAoK $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ and obtains an output w. Obviously the extractor is applied to an augmented machine in order to consider also the remaining part of the execution of P_π^* with $E_\pi^{\mathtt{rand}}$. Finally $E_\pi^{\mathtt{rand}}$ outputs pairs (x, w) corresponding respectively to the instances successfully proved by P_π^* during the execution with $V_\pi^{\mathtt{rand}}$ and the associated $\mathcal{NP}$ witnesses obtained by the extractor associated to $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$.

Since by construction $E_\pi^{\mathtt{rand}}$ runs the code of $V_\pi^{\mathtt{rand}}$, we have that the first of the two claims of the lemma is satisfied perfectly. The second claim is satisfied unless the extractor of $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ fails. However, by the AoK property of $(P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$, and by using the union bound, we have that the probability that for some x proved to $V_\pi^{\mathtt{rand}}$, the extractor $E_\pi^{\mathtt{rand}}$ fails is negligible.

We finally observe that $E_\pi^{\mathtt{rand}}$ is canonical since it first runs the honest verifier, and then extract from statements proved successfully to the honest verifier.

We can now conclude the proof of resettable soundness, by showing that if P_π^* is successful in proving with non-negligible probability a false instance x to V_π, then we can break the pseudo-random function. Indeed, the difference between V_π and $V_\pi^{\mathtt{rand}}$ lies precisely in the use of a PRF instead of pure randomness. The reduction will be based on two arguments. The first argument is that even in the game with V_π, P_π^* will be straight-line in a sub-session, otherwise it will be immediately clear that from this efficiently observable fact one can construct an adversary that breaks the PRF. It could still be possible that P_π^* proves a false statement when playing with V_π but plays true statements only when playing with $V_\pi^{\mathtt{rand}}$. In this case the above analysis does not reach any contradiction. However we can tackle this with a second argument. The second argument is that on the theorems proved by P_π^* we can apply again the extractor of the $\pi_{\mathtt{rWI}}$ since there always is a straight-line execution of a sub-session. However since the instance is false, the extractor will certainly fail. The failure of the extractor (that instead works fine in the experiment with $V_\pi^{\mathtt{rand}}$) gives again an efficiently observable fact that can be used to construct an adversary that breaks the PRF. We now proceed more formally.

We first prove that the successful PPT P_π^* when playing with V_π and proving a statement it must succeed in at least one sub-session with a given session-prefix in straight line, including the case of a false statement.

Lemma 4 *Assume that V_π accepts for some x the proof given P_π^* with non-negligible probability. Then there is at least one sub-session with a session-prefix q that has been successfully completed on common input x with a transcript $(P_{\mathtt{rWI}1}, V_{\mathtt{rWI}1}, \ldots, P_{\mathtt{rWI}4}, V_{\mathtt{rWI}4})$ and in the whole experiment V_π played in that sub-session with session-prefix q those 4 messages only.*

Proof. Suppose by contradiction that the claim does not hold. We show an adversary $V_\pi^{\mathcal{O}(\cdot)}$ that is successful in breaking the pseudorandom function PRF distinguishing whether the oracle $\mathcal{O}(\cdot)$ implements a random function or a PRF. Notice that the difference between V_π and $V_\pi^{\mathtt{rand}}$ consists in the use of a PRF in the former and of pure randomness in the latter. However observe that the reset attack of P_π^* consists possibly in running the verifier with $s(k)$ different randomnesses, also rewinding each execution with a given randomness. Therefore we have that V_π has a different randomness that will be used as seed in different PRFs. We will use hybrid arguments in order to concentrate on one PRF only. We will use the word "incarnation" to mean an execution on a given randomness of the verifier selected among the $s(k)$ available randomnesses.

Let $V_\pi^{i,\mathcal{O}(\cdot)}$ be the adversary that will run as V_π in the first i incarnations and as $V_\pi^{\mathtt{rand}}$ in the remaining ones. By definition, $V_\pi^{0,\mathcal{O}(\cdot)}$ is precisely $V_\pi^{\mathtt{rand}}$ and $V_\pi^{s(k),\mathcal{O}(\cdot)}$ is precisely V_π.

The goal of Lemma 4 is to prove that the very same statement proved in Lemma 3 with respect to $V_\pi^{\mathtt{rand}} = V_\pi^{0,\mathcal{O}(\cdot)}$ also holds with respect to $V_\pi = V_\pi^{s(k),\mathcal{O}(\cdot)}$. We therefore proceed by running hybrid arguments. We claim that if that statement holds for $V_\pi^{i,\mathcal{O}(\cdot)}$ then it holds for $V_\pi^{i+1,\mathcal{O}(\cdot)}$. Assume by contradiction that this is not the case. Consider $V_\pi^{\mathcal{O}(\cdot)}$ that runs with P_π^* in the experiment of resettable soundness as follows. It emulates V_π in the first i incarnations, emulates $V_\pi^{\mathtt{rand}}$ in the last $s(n)-(i+1)$ incarnations, while in the $(i+1)$-th incarnation it works as follow: each time V_π queries the PRF with a new input, then the actual output that will be used by $V_\pi^{\mathcal{O}(\cdot)}$ is the output of $\mathcal{O}(\cdot)$. It is clear that when $\mathcal{O}(\cdot)$ is a pure random function then the resulting experiment corresponds to the execution of $V_\pi^{i,\mathcal{O}(\cdot)}$. If instead $\mathcal{O}(\cdot)$ is a PRF, then the resulting experiment corresponds to the execution of $V_\pi^{i+1,\mathcal{O}(\cdot)}$. Therefore by the security of the PRF, and the fact that in the execution with $V_\pi^{i,\mathcal{O}(\cdot)}$ there is at least a sub-session that is solved in straight-line, we have that the same holds with $V_\pi^{i+1,\mathcal{O}(\cdot)}$. Therefore the claim holds for $V_\pi = V_\pi^{s(k),\mathcal{O}(\cdot)}$.

We have proved that a PPT adversary P_π^* can succeed in proving an instance to V_π only if there is at least a given session-prefix so that the corresponding sub-session is solved in straight line.

Similarly to $E_\pi^{\mathtt{rand}}$, consider now the *canonical* extractor $E_\pi^{\mathtt{prf}}$ that simply runs the code of V_π with P_π^*. During the interaction, for every instance x successfully proved by P_π^*, $E_\pi^{\mathtt{prf}}$ stores the session-prefixes of the straight-line sub-sessions. Next, for each such session-prefix, sequentially, $E_\pi^{\mathtt{prf}}$ runs the extractor $E_{\mathtt{rWI}}$ associated to $\pi_{\mathtt{rWI}}$. Obviously the extractor is applied to an augmented machine in order to consider also the remaining part of the execution of P_π^* with $E_\pi^{\mathtt{prf}}$. Clearly, we have that the extractor $E_{\mathtt{rWI}}$ will always fail since the statement is false and thus there is no witness that can be extracted.

We can therefore extend the code of the adversary $V_\pi^{\mathcal{O}(\cdot)}$ discussed in the proof of Lemma 4, so that once the straight-line executions are known, it can apply the extractor $E_{\mathtt{rWI}}$ associated to $\pi_{\mathtt{rWI}}$ one-by-one to all proved statements and see whether it is always successful or not. Notice that when $\mathcal{O}(\cdot)$ is running a random function, we have that $V_\pi^{\mathcal{O}(\cdot)}$ is behaving identically to $V_\pi^{i,\mathcal{O}(\cdot)}$, and thus by Lemma 2 and Lemma 3 the witnesses are successfully extracted. Instead, when $\mathcal{O}(\cdot)$ runs a pseudorandom function, we have that $V_\pi^{\mathcal{O}(\cdot)}$ is behaving identically to $V_\pi^{i+1,\mathcal{O}(\cdot)}$ with a follow up use of $E_{\mathtt{rWI}}$, and we have just proved that in this case extraction can not be always successful. Therefore $V_\pi^{\mathcal{O}(\cdot)}$ can distinguish the actual function used by $\mathcal{O}(\cdot)$. This is a contradiction that concludes the proof of Theorem 1.

5 7-Round Resettable WI

In this section we show a 7-round rWIAoK for $\mathcal{NP}$.

We start by recalling in Fig. 2 the relation $\mathbf{R}_{\text{sim}}$ introduced by Barak [Bar01, Bar04]. We also consider the construction given by Pass and Rosen in [PR05b, PR08] that achieved public-coin constant-round statistical ZKAoK from CRHFs, on top of Barak's techniques.

In our notation, $\mathbf{SHCom}_{h_{com}}$ is the 2nd round of a statistically hiding commitment scheme, where the first round consists in selecting a function h_{com} from family of CRHFs $\mathcal{H}_k$ mapping $\{0,1\}^*$ to $\{0,1\}^k$.

Fig. 2 shows an oversimplified relation that would base the security of our protocol on collision-resistant against "slightly" superpolynomial-sized circuits. As discussed in [BG02, Bar04, PR05b], in order to prove security by relying on standard CRHFs, the actual relation should also include an error-correcting code $\mathtt{ECC}$ (with constant distance and with polynomial-time encoding and decoding) and the condition $c = \mathbf{SHCom}_{h_{com}}(h(\Pi), s)$ should be replaced by $c = \mathbf{SHCom}_{h_{com}}(h(\mathtt{ECC}(\Pi)), s)$. However in order to simplify the notation we will omit this from our discussion.

Instance: $(h_{com}, h, c, r) \in \mathcal{H}_k \times \mathcal{H}_k \times \{0,1\}^k \times \{0,1\}^{\mathtt{poly(k)}}$.
Witness: $\Pi \in \{0,1\}^*$, $y \in \{0,1\}^*$ and $s \in \{0,1\}^{3k}$.
Relation: $\mathbf{R}_{\text{sim}}((h_{com}, h, c, r), (\Pi, y, s)) = 1$ if and only if :

1. $|y| \leq |r| - k$.
2. $c = \mathbf{SHCom}_{h_{com}}(h(\Pi), s)$.
3. $\Pi(y) = r$ within $T(n)$ steps.

Figure 2: The $\mathbf{NTIME}(T(k))$ relation $\mathbf{R}_{\text{sim}}$.

In Fig. 3 we show the special universal argument $\pi_{\text{sUA}} = (P_{\text{sUA}}, V_{\text{sUA}})$ presented in [PR05b]. It consists of two parts. First, P_{sUA} and V_{sUA} play the 4-round public-coin universal argument $\pi_{\text{UA}} = (P_{\text{UA}}, V_{\text{UA}})$ of Barak and Goldreich [BG02] with the following change: the two messages of P_{UA} are committed through a statistically hiding commitment scheme. We denote such 4 rounds as $\mathtt{sUA}_1, \ldots, \mathtt{sUA}_4$. We stress here that $\mathtt{sUA}_1$ consists in randomly selecting an hash function in a family of CRHFs. Then P_{sUA} and V_{sUA} play a statistical WIAoK where P_{sUA} proves knowledge of either $x \in L$ or of openings of the two above commitments so that the 4 messages of the underlying universal argument π_{UA} would be accepted by V_{UA}. This is typically achieved by using the 3-round WI proof of knowledge (PoK, for short) for Hamiltonicity of Blum [Blu86] $\pi_{\mathtt{BL}} = (P_{\mathtt{BL}}, V_{\mathtt{BL}})$, in a 4-round implementation so that first the verifier sends h_{com} and then the 3 rounds are played with $P_{\mathtt{BL}}$ that commits by means of $\mathbf{SHCom}_{h_{com}}$. We denote such a 4-round WIAoK as $\pi_{\mathtt{sWI}} = (P_{\mathtt{sWI}}, V_{\mathtt{sWI}})$ and in Fig. 3 we name the corresponding 4 messages as $\mathtt{sUA}_5, \ldots, \mathtt{sUA}_8$. This special universal argument is proven in [PR05b, PR08] to be statistical WI and is beneficial to then obtain a constant-round public-coin statistical ZKAoK.

Common Input: $x \in \{0,1\}^k$, an instance (h_{com}, h, c, r) for $\mathbf{R}_{\text{sim}}$.

Stage 1 (Encrypted UARG):

$\mathtt{sUA}_1 : V_{\text{sUA}} \to P_{\text{sUA}}$: Send $\alpha \xleftarrow{R} \{0,1\}^k$.
$\mathtt{sUA}_2 : P_{\text{sUA}} \to V_{\text{sUA}}$: Send $\hat{\beta} = \mathbf{SHCom}_{h_{com}}(0^k, s_0)$.
$\mathtt{sUA}_3 : V_{\text{sUA}} \to P_{\text{sUA}}$: Send $\gamma \xleftarrow{R} \{0,1\}^k$.
$\mathtt{sUA}_4 : P_{\text{sUA}} \to V_{\text{sUA}}$: Send $\hat{\delta} = \mathbf{SHCom}_{h_{com}}(0^k, s_1)$.

Stage 2 (Body of the proof):

$\mathtt{sUA}_5, \mathtt{sUA}_6, \mathtt{sUA}_7, \mathtt{sUA}_8 : P_{\text{sUA}} \Leftrightarrow V_{\text{sUA}}$: 4-round statistical WIAoK $(P_{\mathtt{sWI}}, V_{\mathtt{sWI}})$ proving the OR of the following statements:

1. $\exists w \in \{0,1\}^{\mathtt{poly(|x|)}}$ s.t. $R_L(x, w) = 1$.
2. $\exists(\beta, \delta, s_1, s_2)$ s.t. $\hat{\beta} = \mathbf{SHCom}_{h_{com}}(\beta, s_1)$, $\hat{\delta} = \mathbf{SHCom}_{h_{com}}(\delta, s_2)$, $(\alpha, \beta, \gamma, \delta)$ is an accepting transcript for $(P_{\text{UA}}, V_{\text{UA}})$ proving statement $\mathbf{R}_{\text{sim}}((h_{com}, h, c, r), (\Pi, y, s)) = 1$.

Figure 3: Statistical UA $\pi_{\text{sUA}} = (P_{\text{sUA}}, V_{\text{sUA}})$.

In Fig. 4 we show the construction[5] of Pass and Rosen [PR05b, PR08] of a public-coin constant-round statistical ZKAoK for $\mathcal{NP}$. It consists of using Barak's protocol with the exception of committing through a statistically hiding commitment scheme, and of using the special universal argument described in Fig. 3. Putting all pieces together, the total round complexity is 10.

We now show our 6-round public-coin statistical ZKAoK $\pi'_{\mathtt{zk}} = (P'_{\mathtt{zk}}, V'_{\mathtt{zk}})$ that only consist

[5] We actually show a simplification of their construction since they also implemented a two-slot technique in order to achieve non-malleability.

Common Input: $x \in \{0,1\}^k$.

Stage 0 (Set-up):
$V_{\mathtt{zk}} \to P_{\mathtt{zk}}$: Send $h \stackrel{R}{\leftarrow} \mathcal{H}_K$, $h_{com} \stackrel{R}{\leftarrow} \mathcal{H}_K$.
Stage 1 (Preamble):
$P_{\mathtt{zk}} \to V_{\mathtt{zk}}$: Send $c = \mathbf{SHCom}_{h_{com}}(0^k, s)$.
$V_{\mathtt{zk}} \to P_{\mathtt{zk}}$: Send $r \stackrel{R}{\leftarrow} \{0,1\}^{3k}$.
Stage 2 (Body of the proof):
$P_{\mathtt{zk}} \Leftrightarrow V_{\mathtt{zk}}$: special universal argument $(P_{\mathrm{sUA}}, V_{\mathrm{sUA}})$ proving the OR of the following statements:
1. $\exists w \in \{0,1\}^{\mathtt{poly}(|\mathtt{x}|)}$ s.t. $R_L(x,w) = 1$.
2. $\exists(\Pi, y, s)$ s.t. $\mathbf{R}_{\mathrm{sim}}((h_{com}, h, c, r), (\Pi, y, s)) = 1$.

Figure 4: ZKAoK $\pi_{\mathtt{zk}} = (P_{\mathtt{zk}}, V_{\mathtt{zk}})$.

in implementing differently the protocol of Fig. 4, making use of the Hamiltonicity proof system due to Lapidot and Shamir [LS90]. The special property of the proof system of [LS90] is that the actual statement to be proved and its witness are needed only when playing the last round. For more details, see Appendix B.

Previous rounds only need the size of the statement. It can be implemented as a 4-round statistical WIAoK, where the first round as for Blum's protocol, just consists in selecting the function from the family of CRHFs in order to let the prover sends a statistically hiding commitment. We will denote by $\mathtt{LS}_1, \ldots, \mathtt{LS}_4$ the four rounds of such a special statistical WIAoK, and it will be used to prove the same statement proved in Stage 2 of Fig. 3.

Another crucial property of our $\pi'_{\mathtt{zk}}$ is that the actual statement to be proven needs to be known only when playing the 4-th round (i.e., $\mathtt{ZK}_4$ in Fig. 5), while the first 3 rounds only need the size of the statement.

The protocol is depicted in Fig. 5.

Common Input: $x \in \{0,1\}^k$.

$\mathtt{ZK}_1 : V'_{\mathtt{zk}} \to P'_{\mathtt{zk}}$: Send $h \stackrel{R}{\leftarrow} \mathcal{H}_K$, $h_{com} \stackrel{R}{\leftarrow} \mathcal{H}_K$.
$\mathtt{ZK}_2 : P'_{\mathtt{zk}} \to V'_{\mathtt{zk}}$: Send $c = \mathbf{SHCom}_{h_{com}}(0^k, s)$.
$\mathtt{ZK}_3 : V'_{\mathtt{zk}} \to P'_{\mathtt{zk}}$: Send $r \stackrel{R}{\leftarrow} \{0,1\}^{3k}$, run $\mathtt{sUA}_1, \mathtt{LS}_1$.
$\mathtt{ZK}_4 : P'_{\mathtt{zk}} \to V'_{\mathtt{zk}}$: Run $\mathtt{sUA}_2, \mathtt{LS}_2$.
$\mathtt{ZK}_5 : V'_{\mathtt{zk}} \to P'_{\mathtt{zk}}$: Run $\mathtt{sUA}_3, \mathtt{LS}_3$.
$\mathtt{ZK}_6 : P'_{\mathtt{zk}} \to V'_{\mathtt{zk}}$: Run $\mathtt{sUA}_4, \mathtt{LS}_4$.

Figure 5: 6-Round ZKAoK $\pi'_{\mathtt{zk}} = (P'_{\mathtt{zk}}, V'_{\mathtt{zk}})$.

Lemma 5 *Assuming the existence of a family of CRHFs, there exists a* 6*-round public-coin statistical ZKAoK for* $\mathcal{NP}$*. Moreover the first* 3 *rounds only need the size of the statement.*

Proof. Consider the protocol depicted in Fig 5. First of all, notice that all ingredients can be constructed by only relying on CRHFs. Then, notice that the protocol is just a different implementation of the tools used by the protocol of Pass and Rosen shown in Fig. 4, with the exception of playing subprotocols in parallel rather then sequentially. However, by inspection each round of the protocol can be played by all involved parties (i.e., prover, verifier, simulator,

extractor) including adversaries to be used in the reductions of security proofs. All the analysis of [PR05a] applies untouched. Therefore, the claim holds.

Lemma 6 *Assuming the existence of a family of CRHFs, there exists a* 6*-round rsZKAoK for* $\mathcal{NP}$*. Moreover the first* 3 *rounds only need the size of the statement.*

Proof. By observing that the transformation of [BGGL01] on input $\pi'_{\mathtt{zk}}$ will produce the desired 6-round rsZKAoK for $\mathcal{NP}$ we have that the claim holds.

The above Lemma 6 will now be used to construct a 7-round rWIAoK $\pi_{\mathtt{rWI}} = (P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$ for $\mathcal{NP}$. The construction can be found in Fig. 6. Let $\pi''_{\mathtt{zk}} = (P''_{\pi}, V''_{\pi})$ be the 6-round rsZKAoK obtained from Lemma 6, and let $\mathtt{rsZK}_1, \ldots, \mathtt{rsZK}_6$ be the corresponding 6 rounds.

$\pi_{\mathtt{rWI}}$ will use $\pi''_{\mathtt{zk}}$ and a special implementation of Blum's Hamiltonicity protocol for the statement "$x \in L$" that we describe now. Blum's protocol is a 3-round WIPoK for $\mathcal{NP}$, and the 3 rounds are computed as follows: 1) the prover commits to k permuted adjacency matrixes; 2) the verifier sends a k-bit random string; 3) the prover opens some of the committed bits and shows some permutations according to the received k-bit string. A central idea of our construction is to use the technique of [CGGM00] to make Blum's protocol a rWI proof, and then the technique of [BGGL01] to make it an argument of knowledge. More in details, following the paradigm of [CGGM00], we ask the verifier $V_{\mathtt{rWI}}$ to commit, by means of a statistically hiding commitment, to the k-bit challenge. This commitment is sent before the prover $P_{\mathtt{rWI}}$ commits to permuted adjacency matrixes. The commitment of the k-bit challenge will be opened by $V_{\mathtt{rWI}}$ after that $P_{\mathtt{rWI}}$ commits to permuted adjacency matrixes. Moreover $P_{\mathtt{rWI}}$ will use a PRF on input the commitment of the challenge sent by $V_{\mathtt{rWI}}$ in order to establish the randomness to be used in the next message. This is the paradigm of [CGGM00] that transforms Blum's protocol making it secure against resets of the adversarial verifier. Indeed as proved already in [CGGM00], the resulting protocol is a rWI proof system. The statistically hiding commitment of the challenge will add two rounds to the original 3 rounds of Blum's protocol. Let us denote by $\mathtt{BL}_1, \ldots, \mathtt{BL}_5$ the resulting 5 rounds of this rWI proof system. We then use the technique of [BGGL01] to make this protocol an argument of knowledge with a (necessarily) non-black-box extractor. Indeed we change $\mathtt{BL}_4$ and instead of sending the challenge and the decommitment information corresponding to the commitment played in $\mathtt{BL}_2$, we will ask $V_{\mathtt{rWI}}$ to send the challenge only. Let us assume that in the original round $\mathtt{BL}_4$ the message sent consisted of a pair (u, v) corresponding to challenge and decommitment information. We define $\mathtt{BL}'_4$ as u only, while v will be part of the witness to be used in a statistical rsZKAoK where the verifier $V_{\mathtt{rWI}}$ proves knowledge of the decommitment information v. We implement the rsZKAoK with $\pi''_{\mathtt{zk}}$, and we denote by $\mathtt{rsZK}_1, \ldots, \mathtt{rsZK}_6$ the corresponding messages.

Common Input: $x \in \{0,1\}^k$.

$P_{\mathtt{rWI}} \to V_{\mathtt{rWI}}$: Run $\mathtt{BL}_1, \mathtt{rsZK}_1$.
$V_{\mathtt{rWI}} \to P_{\mathtt{rWI}}$: Run $\mathtt{BL}_2, \mathtt{rsZK}_2$.
$P_{\mathtt{rWI}} \to V_{\mathtt{rWI}}$: Run $\mathtt{BL}_3, \mathtt{rsZK}_3$.
$V_{\mathtt{rWI}} \to V'_{\mathtt{rWI}}$: Run $\mathtt{BL}'_4, \mathtt{rsZK}_4$.
$P_{\mathtt{rWI}} \to P'_{\mathtt{rWI}}$: Run $\mathtt{rsZK}_5$.
$V_{\mathtt{rWI}} \to V'_{\mathtt{rWI}}$: Run $\mathtt{rsZK}_6$.
$P_{\mathtt{rWI}} \to P'_{\mathtt{rWI}}$: Run $\mathtt{BL}_5$.

Figure 6: 7-round rWIAoK $\pi_{\mathtt{rWI}} = (P_{\mathtt{rWI}}, V_{\mathtt{rWI}})$.

Theorem 2 *Assuming the existence of a family of CRHFs, there exists a* 7*-round rWIAoK for* $\mathcal{NP}$*.*

Proof. The proof of this theorem is by inspection. Indeed our protocol is only a special implementation (achieving a better round complexity) of the protocol already given in [BGGL01].

6 Simultaneously Resettable ZK

In this section we prove the main theorem of this work, namely: CRHFs imply the existence of a rsrZKAoK for $\mathcal{NP}$. Therefore not only we go beyond the BGGL conjecture, (surprisingly removing the need of ZAPs and thus highly improving on the computational assumptions for simultaneously resettable zero knowledge), but we also give an argument of knowledge. This is important both for applications that rely on knowledge of the witness (e.g., identification schemes) and to prove security of larger protocols that will use our rsrZKAoK as subprotocol.

We rely on the rsrZK argument system of [DGS09]. In the following discussion we will however refer to their full version [GS08].

By inspection, it is immediate to see that the security proofs of the construction of [GS08] when referring to a ZAP[6] relies only on the fact that a ZAP is a rsrWI proof (i.e., unconditionally sound) system. Therefore, any (even interactive) rsrWI proof system can replace the ZAP in [DGS09] to obtain a rsrZK argument. Still we can not just replace the ZAP with a rsrWIAoK because the security proof of [DGS09] uses the unconditional soundness of the ZAP. Therefore while a rsrWI proof system can safely replace a ZAP, the use of a rsrWIAoK might require some changes. Fortunately we can show that the protocol of [DGS09] is still a rsrZK argument when a rsrWIAoK is used instead of ZAPs. Indeed, we show that whatever was proved in [DGS09, GS08] relying on the unconditional security of the ZAP can be proved here by using the computational resettable soundness property of our rsrWIAoK.

Theorem 3 of [GS08]. In the proof of Theorem 3 of [GS08] (see last part of page 17), the soundness of their construction is proved in a hybrid model. Their proof assumes by contradiction that an adversarial prover P^* can prove a false statement "$x \in L$" with noticeable probability ϵ. In turn, this implies that P^* will successfully complete a ZAP proving "$x \in L \vee trap = Com(1)$". They show that it is still noticeable the probability that P^* manages to convince the verifier for the same false statement "$x \in L$" even when the verifier sets $trap = Com(0)$. This means that P^* is proving with noticeable probability a false statement "$x \in L \vee trap = Com(1)$". They conclude that this contradicts the resettable soundness of the ZAP.

Notice that when the ZAP is replaced by a rsrWIAoK $\pi_{\mathtt{rWI}}$ there is no issue in the above proof. Indeed, instead of relying on unconditional resettable soundness, it is standard to show a reduction that reduces P^* to an adversary $P^*_{\mathtt{rWI}}$ of $\pi_{\mathtt{rWI}}$.

Lemma 3 of [GS08]. Experiment 2 in the proof of Lemma 3 of [GS08], similarly to Theorem 3 above, uses the unconditional resettable soundness of the ZAP. They claim that V^* can not deviate from the committed randomness (property 3 of a prover-admissible proof system) since V^* has to prove with a ZAP that either the played messages are consistent with the committed randomness or $trap = Com(1)$. Since it is already known (by a resettably-sound ZK argument previously given by the verifier) that $trap = Com(0)$, we have that the statement of the ZAP is false. Therefore the unconditional soundness of the ZAP guarantees that the verifier did not deviate from the committed randomness.

[6] For simplicity here we just say ZAP to refer to the implementation of a ZAP that is resettably secure. In [DGS09, GS08] this is named rZAP.

Again, when the ZAP is replaced by a rsrWIAoK there is no issue in the above proof. Indeed, instead of relying on unconditional resettable soundness, it is standard to show a reduction that reduces P^* to an adversary $P^*_{\mathtt{rWI}}$.

Theorem 5 of [GS08]. This theorem is the dual of the above Lemma 3. Indeed it is now a successful adversarial prover P^* that gives a ZAP proving consistency of the played messages with respect to the committed randomness or that "$x \in L$". Since "$x \in L$" must be false[7], we have by the unconditional soundness of the ZAP that P^* can not deviate from the committed randomness.

Again, as explained in the previous cases, the proof goes through also when the ZAP is replaced by a rsrWIAoK.

The formal claim. The previous discussion shows that the argument system of [DGS09] is still secure under simultaneous reset attacks when our rsrWIAoK $\pi_{\mathtt{rWI}}$ replaces their ZAPs. We finally notice that their protocol ended with a ZAP proving "$x \in L \vee trap = Com(1)$". By replacing this ZAP with a rsrWIAoK we have that the resulting argument system is an argument of knowledge. Indeed the extractor $E_{\mathtt{rWI}}$ can be applied to the augmented machine consisting of P^* and the honest verifier behavior with the exception of this run of $\pi_{\mathtt{rWI}}$ from which extraction is desired.

Putting all pieces together, we have proven the following claim.

Theorem 3 *If there exists a family of CRHFs, then there exists (constructively) a rsrZKAoK for $\mathcal{NP}$.*

Acknowledgments

We thank Shafi Goldwasser and Alessandra Scafuro for valuable discussions.

Research supported in part by NSF grants CCF-0916574; IIS-1065276; CCF-1016540; CNS-1118126; CNS-1136174; US-Israel BSF grant 2008411, OKAWA Foundation Research Award, IBM Faculty Research Award, Xerox Faculty Research Award, B. John Garrick Foundation Award, Teradata Research Award, and Lockheed-Martin Corporation Research Award, and European Commission through the FP7 programme under contract 216676 ECRYPT II. This material is also based upon work supported by the Defense Advanced Research Projects Agency through the U.S. Office of Naval Research under Contract N00014-11-1-0392. The views expressed are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

References

[APV05] Joël Alwen, Giuseppe Persiano, and Ivan Visconti. Impossibility and feasibility results for zero knowledge with public keys. In *Advances in Cryptology - CRYPTO 2005: 25th Annual International Cryptology Conference, Santa Barbara, California, USA, August 14-18, 2005, Proceedings*, volume 3621 of *Lecture Notes in Computer Science*, pages 135–151. Springer, 2005.

[Bar01] Boaz Barak. How to go beyond the black-box simulation barrier. In *FOCS*, pages 106–115, 2001.

[Bar04] Boaz Barak. Non-black-box techniques in cryptography. PhD Thesis, 2004.

[7] Recall that in this part of the proof we are assuming that P^* is successful and therefore "$x \in L$" is false.

[BFGM01] Mihir Bellare, Marc Fischlin, Shafi Goldwasser, and Silvio Micali. Identification protocols secure against reset attacks. In *EUROCRYPT*, LNCS, pages 495–511, 2001.

[BG02] Boaz Barak and Oded Goldreich. Universal arguments and their applications. In *CCC*, pages 194–203, 2002.

[BG08] Boaz Barak and Oded Goldreich. Universal arguments and their applications. *SIAM J. Comput.*, 38(5):1661–1694, 2008.

[BGGL01] Boaz Barak, Oded Goldreich, Shafi Goldwasser, and Yehuda Lindell. Resettably-sound zero-knowledge and its applications. In *FOCS*, pages 116–125. IEEE Computer Society, 2001.

[Blu86] Manuel Blum. How to Prove a Theorem So No One Else Can Claim I t. In *Proceedings of the International Congress of Mathematicians*, pages 1444–1451, 1986.

[BLV03] Boaz Barak, Yehuda Lindell, and Salil P. Vadhan. Lower bounds for non-black-box zero knowledge. In *FOCS*, pages 384–393. IEEE Computer Society, 2003.

[BLV06] Boaz Barak, Yehuda Lindell, and Salil P. Vadhan. Lower bounds for non-black-box zero knowledge. *J. Comput. Syst. Sci.*, 72(2):321–391, 2006.

[BOV12] Joshua Baron, Rafail Ostrovsky, and Ivan Visconti. Nearly simultaneously resettable black-box zero knowledge. In *Automata, Languages, and Programming - 39th International Colloquium, ICALP 2012, Warwick, UK, July 9-13, 2012, Proceedings, Part I*, volume 7391 of *Lecture Notes in Computer Science*, pages 88–99. Springer, 2012.

[BP12] Nir Bitansky and Omer Paneth. From the impossibility of obfuscation to a new non-black-box simulation technique. In *FOCS*, pages –, 2012.

[BPSV08] Carlo Blundo, Giuseppe Persiano, Ahmad-Reza Sadeghi, and Ivan Visconti. Improved security notions and protocols for non-transferable identification. In *Computer Security - ESORICS 2008, 13th European Symposium on Research in Computer Security, Málaga, Spain, October 6-8, 2008. Proceedings*, volume 5283 of *Lecture Notes in Computer Science*, pages 364–378. Springer, 2008.

[CGGM00] Ran Canetti, Oded Goldreich, Shafi Goldwasser, and Silvio Micali. Resettable zero-knowledge (extended abstract). In *STOC*, pages 235–244, 2000.

[CKPR01] Ran Canetti, Joe Kilian, Erez Petrank, and Alon Rosen. Black-box concurrent zero-knowledge requires omega∼(log n) rounds. In *STOC*, pages 570–579. ACM, 2001.

[COSV12] Chongwon Cho, Rafail Ostrovsky, Alessandra Scafuro, and Ivan Visconti. Simultaneously resettable arguments of knowledge. In *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 530–547. Springer, 2012.

[CPV04a] Giovanni Di Crescenzo, Giuseppe Persiano, and Ivan Visconti. Constant-round resettable zero knowledge with concurrent soundness in the bare public-key model. In *Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19, 2004, Proceedings*, volume 3152 of *Lecture Notes in Computer Science*, pages 237–253. Springer, 2004.

[CPV04b] Giovanni Di Crescenzo, Giuseppe Persiano, and Ivan Visconti. Improved setup assumptions for 3-round resettable zero knowledge. In *Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings*, volume 3329 of *Lecture Notes in Computer Science*, pages 530–544. Springer, 2004.

[DFG+11] Yi Deng, Dengguo Feng, Vipul Goyal, Dongdai Lin, Amit Sahai, and Moti Yung. Resettable cryptography in constant rounds - the case of zero knowledge. In *ASIACRYPT*, 2011.

[DGS09] Yi Deng, Vipul Goyal, and Amit Sahai. Resolving the simultaneous resettability conjecture and a new non-black-box simulation strategy. In *FOCS*, pages 251–260. IEEE Computer Society, 2009.

[DL07] Yi Deng and Dongdai Lin. Instance-dependent verifiable random functions and their application to simultaneous resettability. In *EUROCRYPT*, pages 148–168, 2007.

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

[DN07] Cynthia Dwork and Moni Naor. Zaps and their applications. *SIAM J. Comput.*, 36(6):1513–1543, 2007.

[DNS98] Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *STOC '98*, pages 409–418. ACM, 1998.

[FLS90] Uriel Feige, Dror Lapidot, and Adi Shamir. Multiple non-interactive zero knowledge proofs based on a single random string (extended abstract). In *FOCS*, pages 308–317. IEEE Computer Society, 1990.

[GKOV12] Sanjam Garg, Abishek Kumarasubramanian, Rafail Ostrovsky, and Ivan Visconti. Impossibility results for static input secure computation. In *CRYPTO*, volume 7417 of *Lecture Notes in Computer Science*, pages 424–442. Springer, 2012.

[GM11] Vipul Goyal and Hemanta K. Maji. Stateless cryptographic protocols. In *FOCS*, pages 678–687. IEEE, 2011.

[GMR85] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof-systems. In *STOC '85*, pages 291–304. ACM, 1985.

[Gol11] Oded Goldreich. Basing non-interactive zero-knowledge on (enhanced) trapdoor permutations: The state of the art. In *Studies in Complexity and Cryptography*, volume 6650 of *LNCS*, pages 406–421. Springer, 2011.

[GOS06] Jens Groth, Rafail Ostrovsky, and Amit Sahai. Non-interactive zaps and new techniques for nizk. In *Advances in Cryptology – CRYPTO 06*, volume 4117 of *Lecture Notes in Computer Science*, pages 97–111. Springer, 2006.

[GOVW12] Sanjam Garg, Rafail Ostrovsky, Ivan Visconti, and Akshay Wadia. Resettable statistical zero knowledge. In *TCC*, volume 7194 of *Lecture Notes in Computer Science*, pages 494–511. Springer, 2012.

[GS08] Vipul Goyal and Amit Sahai. Resolving the simultaneous resettability conjecture and a new non-black-box simulation strategy. cryptology eprint archive, report 2008/545, 2008. `http://eprint.iacr.org/`.

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

[MR01a] Silvio Micali and Leonid Reyzin. Min-round resettable zero-knowledge in the public-key model. In *EUROCRYPT*, volume 2045 of *Lecture Notes in Computer Science*, pages 373–393. Springer-Verlag, 2001.

[MR01b] Silvio Micali and Leonid Reyzin. Soundness in the public-key model. In *Advances in Cryptology – Crypto '01*, volume 2139 of *Lecture Notes in Computer Science*, pages 542–565. Springer-Verlag, 2001.

[PR05a] Rafael Pass and Alon Rosen. Concurrent non-malleable commitments. In *In 46th Annual Symposium on Foundations of Computer Science*, FOCS '05, pages 563–572. IEEE Computer Society Press, 2005.

[PR05b] Rafael Pass and Alon Rosen. New and improved constructions of non-malleable cryptographic protocols. In *STOC '05*, pages 533–542. ACM, 2005.

[PR08] Rafael Pass and Alon Rosen. New and improved constructions of nonmalleable cryptographic protocols. *SIAM J. Comput.*, 38(2):702–752, 2008.

[Ros00] Alon Rosen. A note on the round-complexity of concurrent zero-knowledge. In *CRYPTO*, volume 1880 of *LNCS*, pages 451–468, 2000.

[SV12] Alessandra Scafuro and Ivan Visconti. On round-optimal zero knowledge in the bare public-key model. In *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 153–171. Springer, 2012.

[VV09] Carmine Ventre and Ivan Visconti. Co-sound zero-knowledge with public keys. In *Progress in Cryptology - AFRICACRYPT 2009, Second International Conference on Cryptology in Africa, Gammarth, Tunisia, June 21-25, 2009. Proceedings*, volume 5580 of *Lecture Notes in Computer Science*, pages 287–304. Springer, 2009.

[Yil10] Scott Yilek. Resettable public-key encryption: How to encrypt on a virtual machine. In *Topics in Cryptology, CT-RSA 2010*, volume 5985 of *Lecture Notes in Computer Science*, pages 41–56. Springer, 2010.

[YZ07] Moti Yung and Yunlei Zhao. Generic and practical resettable zero-knowledge in the bare public-key model. In *EUROCRYPT*, pages 129–147, 2007.

[ZDLZ03] Yunlei Zhao, Xiaotie Deng, Chan H. Lee, and Hong Zhu. Resettable zero-knowledge in the weak public-key model. In *EUROCRYPT*, volume 2045 of *Lecture Notes in Computer Science*, pages 123–139. Springer-Verlag, 2003.

A Definitions

A polynomial-time relation R is a relation for which it is possible to verify in time polynomial in $|x|$ whether $R(x, w) = 1$. Let us consider an $\mathcal{NP}$-language L and denote by R_L the corresponding polynomial-time relation such that $x \in L$ if and only if there exists w such that $R_L(x, w) = 1$. We will call such a w a *valid witness for* $x \in L$. A *negligible* function $\nu(k)$ is a non-negative function such that for any constant $c < 0$ and for all sufficiently large k, $\nu(k) < k^c$. We will call a positive function *overwhelming* if it can be described as $1 - \nu$ for some negligible function nu. We will denote by $\text{Prob}_r[\ X\ ]$ the probability of an event X over coins r.

Indistinguishability.

Definition 1 *Two ensembles of distributions $X = \{X_k\}$ and $Y = \{Y_k\}$ ranging over $\{0,1\}^{\mathtt{poly(k)}}$ are* computationally indistinguishable *if for any polynomial-sized circuit D there exists a negligible function ν such that*

$$\left|\text{Prob}[\ \alpha \leftarrow X_s : D(s,\alpha) = 1\ ] - \text{Prob}[\ \alpha \leftarrow Y_s : D(s,\alpha) = 1\ ]\right| < \nu(k).$$

A.1 Commitment Schemes

We now give a definition for several statistically hiding commitment schemes. For readability we will use "for all m" to mean any possible message m of length polynomial in the security parameter.

Definition 2 $(\mathsf{Gen}, \mathbf{SHCom}_{h_{com}}, \mathbf{SHVer}_{h_{com}})$ *is a* statistically hiding commitment scheme *if:*

- **efficiency:** Gen, $\mathbf{SHCom}_{h_{com}}$ *and* $\mathbf{SHVer}_{h_{com}}$ *are polynomial-time algorithms;*
- **completeness:** *for all m it holds that* $\text{Prob}[h_{com} \leftarrow \mathsf{Gen}(1^k); (\text{COM}, \mathsf{dec}) \leftarrow \mathbf{SHCom}_{h_{com}}(h_{com}, m) : \mathbf{SHVer}_{h_{com}}(h_{com}, \text{COM}, \mathsf{dec}, m) = 1] = 1$;
- **binding:** *for any polynomial-time algorithm* $\mathtt{committer}^*$ *there is a negligible function ν such that for all sufficiently large k it holds that:*

  $\text{Prob}[h_{com} \leftarrow \mathsf{Gen}(1^k); (\text{COM}, m_0, m_1, \mathsf{dec}_0, \mathsf{dec}_1) \leftarrow \mathtt{committer}^*(h_{com}) :$

  $m_0 \neq m_1$ and $\mathbf{SHVer}_{h_{com}}(h_{com}, \text{COM}, \mathsf{dec}_0, m_0) = \mathbf{SHVer}_{h_{com}}(h_{com}, \text{COM}, \mathsf{dec}_1, m_1) = 1] \leq \nu(k)$;
- **hiding:** *for any algorithm* $\mathtt{receiver}^*$ *there is a negligible function ν such that for all m_0, m_1 where $|m_0| = |m_1|$ and all sufficiently large k it holds that* $\text{Prob}\Big[(h_{com}, \mathtt{aux}) \leftarrow \mathtt{receiver}(1^k); b \leftarrow \{0,1\}; (\text{COM}, \mathsf{dec}) \leftarrow \mathbf{SHCom}_{h_{com}}(h_{com}, m_b) : b \leftarrow \mathtt{receiver}^*(\text{COM}, \mathtt{aux})\Big] \leq \frac{1}{2} + \nu(k)$.

When h_{com} is clear from context, we often say "m, dec is a valid opening for COM*" to mean that* $\mathbf{SHVer}_{h_{com}}(h_{com}, \text{COM}, \mathsf{dec}, m) = 1$.

Collision-resistant hash functions. We will use hash functions as defined below.

Definition 3 *Let $\mathcal{H} = \{h_\alpha\}$ be an efficiently sampleable hash function ensemble where $h_\alpha : \{0,1\}^* \to \{0,1\}^\alpha$. We say that $\mathcal{H}$ is* collision-resistant against polynomial size circuits *if for every (non-uniform) polynomial-size circuit family $\{A_k\}_{k \in N}$, for all positive constants c, and all sufficiently large n, it holds that*

$$\text{Prob}[\ \alpha \xrightarrow{R} \{0,1\}^n : A_k(\alpha) = (x, x') \wedge h_\alpha(x) = h_\alpha(x')\ ] < k^{-c}.$$

Pairwise-independent hash functions. We will make use of a family of pairwise-independent hash functions.

Definition 4 *A family of functions $\mathcal{H} = \{\, h : \{0,1\}^n \to \{0,1\}^m \,\}$ is said to be pairwise-independent iff $\forall x \neq x' \in \{0,1\}^n \, \forall y, y' \in \{0,1\}^m$,*

$$\Pr_{h \leftarrow \mathcal{H}}\left[h(x) = y \wedge h(x') = y'\right] = 2^{-2m}.$$

Theorem 4 *Let F be a finite field. Then the family of functions $\mathcal{H} = h_{a,b} : F \to F_{a,b \in F}$ where $h_{a,b} = ax + b$ is pairwise independent.*

A.2 Proof Systems

An *interactive argument system* for a language L is a pair of probabilistic polynomial time algorithms (P, V), satisfying the requirements of *completeness* and *soundness*. Completeness requires that for any $x \in L$, at the end of the interaction between P and V, V rejects with negligible probability. Soundness requires that for any $x \notin L$, for any probabilistic polynomial time P^*, at the end of the interaction between P^* and V, V accepts with negligible probability. We assume that when V accepts it outputs 1 and 0 otherwise. The output of V when playing with a machine P' on common input x will be denoted by $\langle P', V\rangle(x)$.

Definition 5 *A proof system for the language L, is a pair of interactive Turing machines (P, V) running on common input x such that:*

- *Efficiency: V is* PPT*. When P receives as input an $\mathcal{NP}$ witness w for $x \in L$, then P is* PPT *too.*
- *Completeness: There exists a negligible function $\nu(\cdot)$ such that for every pair (x, w) such that $R_L(x, w) = 1$,*
$$\text{Prob}[\ \langle P(w), V\rangle(x) = 1\ ] \geq 1 - \nu(|x|).$$
- *Soundness: For every $x \notin L$ and for every interactive Turing machine P^* there exists a negligible function $\nu(\cdot)$ such that*
$$\text{Prob}[\ \langle P^*, V\rangle(x) = 1\ ] < \nu(|x|).$$

In the above definition we can relax the soundness requirement by considering P^* as PPT. In this case, we say that (P, V) is an argument system.

We denote by $\texttt{view}^{P(w)}_{V^*(x,z)}$ the view (i.e., its private coins and the received messages) of V^* during an interaction with $P(w)$ on common input x and auxiliary input z.

Zero knowledge. We start with the classical definition of zero knowledge.

Definition 6 *Let (P, V) be an interactive argument system for a language L. We say that (P, V) is zero knowledge if, for any probabilistic polynomial-time adversary V^* receiving an auxiliary input z, there exists a probabilistic polynomial-time algorithm S_{V^*} such for all pairs $(x, w) \in R_L$ the ensembles $\{\texttt{view}^{P(w)}_{V^*(x,z)}\}$ and $\{S_{V^*}(x, z)\}$ are computationally indistinguishable.*

Arguments of knowledge. We use the following variant of the definition of arguments of knowledge, presented in [PR05b].

Definition 7 (Argument of Knowledge) *Let (P, V) be an interactive argument system for the language L with witness relation R_L. We say that (P, V) is an* argument of knowledge *if there exists a polynomial $q(\cdot)$ and a probabilistic oracle machine E, such that for every probabilistic polynomial time interactive machine P^*, for every $x \in L$, and every $y, r \in \{0, 1\}^*$, the following properties hold:*

1. *The expected number of steps taken by E is bounded by*
$$\frac{q(|x|)}{\Pr_\omega[\langle P^*_{x,y,r}, V(x; \omega)\rangle = 1]}$$
2. *The machine E with oracle access to $P^*_{x,y,r}$ outputs a solution $w \in R_L(x)$ with probability at least $1 - \text{negl}(|x|)$.*

Here $P^*_{x,y,r}$ denotes the (deterministic) machine P^* with common input fixed to x, auxiliary input fixed to y, and random tape fixed to r. The machine E is called a (knowledge) extractor*.*

A.3 Resettable Zero Knowledge and Witness Indistinguishability

Definition 8 *Let (P, V) be an interactive proof or argument system for a language L, $t = \mathtt{poly}(\mathtt{k})$, $\bar{x} = x_1, \ldots, x_t$ be a sequence of common inputs and $\bar{w} = w_1, \ldots, w_t$ the corresponding witnesses (i.e., $(x_i, w_i) \in R_L$) for $i = 1, \ldots, t$. We say that a PPT V^* is a* resetting *verifier if it concurrently interacts with an unbounded number of independent copies of P by choosing for each interaction the value i so that the common input will be $x_i \in \bar{x}$, and the prover will use witness w_i, and choosing j so that the prover will use r_j as randomness, with $i, j \in \{1, \ldots, t\}$. The scheduling or the messages to be sent in the different interactions with P are freely decided by V^*. Moreover we say that the transcript of such interactions consist of the common inputs $\bar{x}$ and the sequence of prover and verifier messages exchanged during the interactions. We refer to $\mathtt{view}^{P(\bar{w})}_{V^*(\bar{x},z)}$ as the random variable describing the content of the random tape of V^* and the transcript of the interactions between P and V^*, where z is an auxiliary input received by V^*.*

We use $V^{*P(i,j,y)}$ to denote the next message played by P when running on input (x_i, w_i) and randomness r_j, and when receiving messages $\bar{y} = (y_1, \ldots, y_t)$ (i.e., y_i is the i-th message that V^* sends to P in this given interaction) from a resetting verifier V^*.

Definition 9 *Let (P, V) be an interactive argument system for a language L. We say that $\langle P, V \rangle$ is resettable zero knowledge if, for any PPT resetting verifier V^* there exists a probabilistic polynomial-time algorithm S_{V^*} such that the for all pairs $(\bar{x}, \bar{w}) \in R_L$ the ensembles $\{\mathtt{view}^{P(\bar{w})}_{V^*(\bar{x},z)}\}$ and $\{S_{V^*}(\bar{x}, z)\}$ are computationally indistinguishable.*

Witness indistinguishability. The notion of witness-indistinguishability applies to interactive arguments for $\mathcal{NP}$ languages and requires that no information is revealed to any possibly malicious (but efficient) verifier about which witness is being used during the execution of the argument.

Definition 10 *Let L be a language in $\mathcal{NP}$ and R_L be the corresponding relation. An interactive argument (P, V) for L is* witness indistinguishable *if for every verifier V^*, every pair (w_0, w_1) such that $(x, w_0) \in R_L$ and $(x, w_1) \in R_L$ and every auxiliary input z, the following ensembles are computationally indistinguishable:*

$$\{\mathtt{view}^{P(w_0)}_{V^*(x,z)}\} \quad \textit{and} \quad \{\mathtt{view}^{P(w_1)}_{V^*(x,z)}\}.$$

We now define resettable witness indistinguishability.

Definition 11 *Let L be a language in $\mathcal{NP}$ and R_L be the corresponding relation. An interactive argument $\langle P, V \rangle$ for L is* resettable *witness indistinguishable if for every PPT resetting verifier V^* every $t = \mathtt{poly}(\mathtt{k})$, and every pair $(\bar{w}^0 = (w^0_1, \ldots, w^0_t), \bar{w}^1 = (w^1_1, \ldots, w^1_t))$ such that $(x_i, w^0_i) \in R_L$ and $(x_i, w^1_i) \in R_L$ for $i = 1, \ldots, t$, and any auxiliary input z, the following ensembles are computationally indistinguishable:*

$$\{\mathtt{view}^{P(\bar{w}^0)}_{V^*(\bar{x},z)}\} \quad \textit{and} \quad \{\mathtt{view}^{P(\bar{w}^1)}_{V^*(\bar{x},z)}\}.$$

A construction of a constant-round rWI proof for $\mathcal{NP}$ under the assumption that 2-round perfectly hiding commitment schemes exist is shown in [CGGM00]. It can also be based on CRHFs. In [DN00], a construction of 2-round resettable WI based on NIZK proofs has been shown, and then in [GOS06], a non-interactive rWI proof has been shown by relying on some number-theoretic assumptions.

A.4 Resettably Sound Arguments

Definition 12 ([BGGL01]) *A* resetting attack *of an adversarial prover P^* on a* resettable verifier V^**, is defined by the following two-step random process, indexed by a security parameter n.*

1. *Uniformly select and fix $t = \mathtt{poly(k)}$ random-tapes, denoted $r_1, \ldots, r_t$, for V resulting in deterministic strategies $V^{(j)}(x) = V_{x,r_j}$ defined by $V_{x,r_j}(\alpha) = V(x, r_j, \alpha)$, where $x \in \{0,1\}^n$ and $j \in [t]$. Each $V^{(j)}(x)$ is called an* incarnation *of V.*
2. *On input 1^k, machine P^* is allowed to initiate $\mathtt{poly(k)}$-many interactions with the $V^{(j)}(x)$s. The activity of P^* proceeds in rounds. In each round P^* chooses $x \in \{0,1\}^k$ and $j \in [t]$, thus defining $V^{(j)}(x)$, and conducts a complete session with it.*

Let P and V be some pair of interactive machines, and suppose that V is implementable in probabilistic polynomial-time. We say that (P, V) is a resettably-sound argument system *for L if the following two conditions hold:*

1. Resettable-Completeness: *Consider a polynomial-size resetting attack, and suppose that in some session, after selecting an incarnation $V^{(j)}(x)$, the attacker follows the strategy P. Then, if $x \in L$ then $V^{(j)}(x)$ rejects with negligible probability.*
2. Resettable-Soundness: *For every polynomial-size resetting attack, the probability that in some session the corresponding $V^{(j)}(x)$ has accepted and $x \notin L$ is negligible.*

In [BGGL01], the authors present a transformation that achieves the following result.

Lemma 7 ([BGGL01]) *Let $L \in \mathcal{NP}$ and R_L be a corresponding witness relation. Let (P, V) be a constant-round, public-coin argument of knowledge for R. Then (P, V) can be transformed into (P', V'), such that:*

1. *(P', V') is a resettably-sound argument for L. Furthermore, (P', V') is a resettably-sound argument of knowledge for R_L.*
2. *If (P, V) is witness-indistinguishable, then so is (P', V').*
3. *If (P, V) is zero-knowledge, then so is (P', V').*

Knowledge extraction in the resettably-sound model is defined in [BGGL01] as follows:

Definition 13 *Let $R_L \subseteq \{0,1\}^* \times \{0,1\}^*$ be an $\mathcal{NP}$-relation for an $\mathcal{NP}$-language $L = \{x : \exists w$ such that $(x, w) \in R_L\}$. We say (P, V) is resettably-sound argument of knowledge for R if,*

1. *(P, V) is resettably-sound argument for L, and,*
2. *for every polynomial q, there exists a probabilistic expected polynomial-time oracle machine E such that for every resetting attack P^* of size $q(k)$, the probability that $E^{P^*}(1^k)$ outputs a witness for the input selected in the last session is at most negligibly smaller than the probability that P^* convinces V in the last session.*

A.5 Universal Arguments

A universal argument is an interactive argument of knowledge from proving membership in **Ntime**$(T(k))$ for a super-polynomial function T. We give now the formal definitions, as presented in [BG02].

Definition 14 *Given a description of a Turing machine M, two strings x, w and a number t we say that $(\langle M, x, t\rangle, w) \in R_U$ if M accepts (x, w) within t steps. Moreover, we define $T_M(x, w)$ to be the number of steps made by M on input (x, w).*

Definition 15 *Let $T : N \to N$ be a super-polynomial function. We say that a pair $(\langle M, x, t\rangle, w)$ is in $R_U^{T(k)}$ if $(\langle M, x, t\rangle, w) \in R_U$ and $t \leq T(|\langle M, x, t\rangle|)$. Moreover we define $L_U^{T(k)} = L(R_U^{T(k)})$.*

Definition 16 *A* universal-argument system *is a pair $\langle P, V\rangle$ such that:*

1. **Efficiency:** *V is a probabilistic polynomial-time algorithm.*
2. **Completeness:** *For every $(\langle M, x, t\rangle, w) \in R_u$, $\text{Prob}[\ out_V\langle P(\langle M, x, t\rangle, w), V(\langle M, x, t\rangle)\rangle = 1\ ] = 1$. Moreover, there exists a polynomial p such that $P(w)$, on common input (M, x, t) runs in at most $p(T_M(x, w)) \leq p(t)$ steps.*
3. **Soundness:** *For every polynomial-size circuit family $\{P_k^*\}_{k\in N}$ and every string $y = \langle M, x, t\rangle \in \{0, 1\}^k \backslash L_U$, it holds that $\text{Prob}[\ \texttt{out}_V\langle P^*(y), V(y)\rangle = 1\ ] < \mu(k)$ where $\mu : N \to [0, 1]$ is a negligible function.*
4. **Weak Proof of knowledge:** *For every positive polynomial p there exists a positive polynomial p' and a probabilistic polynomial-time oracle machine E, called the* extractor*, such that the following holds. For every polynomial-size circuit family $\{P'_k\}_{k\in N}$, and every sufficiently long $y = \langle M, x, t\rangle \in \{0, 1\}^*$, if $\text{Prob}[\ \texttt{out}_V\langle P^*(y), V(y)\rangle = 1\ ] \geq 1/p(|y|)$ then*

$$\text{Prob}[\ \exists w = w_1 \cdots w_t \in R_U(y)\ s.t.\ \forall i \in \{1, \ldots, t\}, E_r^{P'_k}(y, i) = w_i\ ] \geq 1/p'(|y|),$$

where $R_U(y)$ is defined as $\{w : (y, w) \in R_U\}$ and $E_r^{P'_k}(\cdot, \cdot)$ denotes the function defined by setting E's random tape equal to r and giving E_r oracle access to P'_k.

We will use the following result by [BG02, BG08], building on techniques in [Bar01]. (A proof for it is divided in [BG02, BG08] among the proofs of Theorem 1.1, Lemma 4.2 and Lemma 4.3.)

Theorem 5 *([BG02, BG08]) Suppose there exists an hash function ensemble that is collision-resistant against polynomial-size circuits. Then there exists a universal argument system with the following property:*

1. *The system is constant-round and public-coin.*
2. *The system is witness indistinguishable.*

Furthermore, for every $\epsilon > 0$, there exists such a system with total communication complexity of m^ϵ, where m is the instance length.

B Special 3-Round WIPoK [LS90]

In the following we describe the 3-round WIPoK protocol for the $\mathcal{NP}$-complete language graph Hamiltonicity ($\mathcal{HC}$), provided by Lapidot and Shamir in [LS90], and we will refer to this construction as `LS` protocol. The reason why this construction is special, is that only the size of the statement need to be known before the last round. The actual statement can therefore be decided during the execution of a larger protocol, and this is very important when one aims at optimizing the overall round complexity.

We now show the protocol assuming that the instance G is known from the beginning, and we discuss later why its knowledge can be postponed to the very last round.

LS protocol consists of k **parallel executions** (with the same input G) of the following protocol:

Inputs: $V_{\mathtt{LS}}$, $P_{\mathtt{LS}}$ have as input a graph G, $P_{\mathtt{LS}}$ has as auxiliary input a witness $w \in R_{\mathcal{HC}}(G)$. Let k be the number of vertexes of G. G is represented by a $k \times k$ adjacency matrix $GMatrix$ where $GMatrix[i][j] = 1$ if there exists an edge between vertexes i and j in G. A non-edge position i, j is a pair of vertexes that are not connected in G and for which $GMatrix[i][j] = 0$.

$\mathtt{LS}_1$ **($P_{\mathtt{LS}} \to V_{\mathtt{LS}}$):** $P_{\mathtt{LS}}$ picks a random k-vertex cycle graph C and commits bit-by-bit to the corresponding adjacency matrix using a statistically binding commitment scheme.

$\mathtt{LS}_2$ **($V_{\mathtt{LS}} \to P_{\mathtt{LS}}$):** $V_{\mathtt{LS}}$ responds with a randomly chosen bit b.

$\mathtt{LS}_3$ **($P_{\mathtt{LS}} \to V_{\mathtt{LS}}$):**

- if $b = 0$, $P_{\mathtt{LS}}$ opens all the commitments, showing that the matrix committed in step $\mathtt{LS}_1$ is actually an k-vertex cycle.
- if $b = 1$, $P_{\mathtt{LS}}$ sends a permutation π mapping the vertex of C in G. Then it opens the commitment of the adjacency matrix of C corresponding to the non-edges of the graph G.
- $V_{\mathtt{LS}}$ accepts if and only if all k sessions are accepting.

LS protocol has the following properties:

WI: The protocol enjoys witness indistinguishability. Indeed, the single execution is zero-knowledge which implies WI and is preserved under parallel and concurrent composition.

Proof of knowledge: Getting the answer for both $b = 0$ and $b = 1$ allows the extraction of the cycle. The reason is the following. For $b = 0$ one gets the random cycle C. Then for $b = 1$ one gets the permutation mapping the random cycle in the actual cycle w that is given to $P_{\mathtt{LS}}$ at the beginning (or before the last message of) the protocol.

Knowledge of statement/witness is required only in Step $\mathtt{LS}_3$: The crucial property is that the first step is independent of the witness and the theorem, since it only requires the sampling of a random k-cycle (k is the size of the theorem and must be known in advance). The witness and theorem are used *only* in the last Step $\mathtt{LS}_3$.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de