Resettable Statistical Zero-Knowledge for NP

Susumu Kiyoshima

NTT Social Informatics Laboratories
susumu.kiyoshima@ntt.com

May 24, 2024

Abstract

Resettable statistical zero-knowledge [Garg–Ostrovsky–Visconti–Wadia, TCC 2012] is a strong privacy notion that guarantees statistical zero-knowledge even when the prover uses the same randomness in multiple proofs.

In this paper, we show an equivalence of resettable statistical zero-knowledge arguments for NP and witness encryption schemes for NP.

- Positive result: For any NP language **L**, a resettable statistical zero-knowledge argument for **L** can be constructed from a witness encryption scheme for **L** under the assumption of the existence of one-way functions.
- Negative result: The existence of even resettable statistical witness-indistinguishable arguments for NP imply the existence of witness encryption schemes for NP under the assumption of the existence of one-way functions.

The positive result is obtained by naturally extending existing techniques (and is likely to be already well-known among experts). The negative result is our main technical contribution.

To explore workarounds for the negative result, we also consider resettable security in a model where the honest party's randomness is only reused with fixed inputs. We show that resettable statistically hiding commitment schemes are impossible even in this model.

This article is a minor revision of the version published by Springer-Verlag available in the proceedings of CRYPTO 2024.

Contents

1 Introduction

Randomness is essential for zero-knowledge proofs/arguments. When either the prover or the verifier is deterministic, zero-knowledge proofs/arguments cannot exist for non-trivial languages [GO94].[1] In contrast, when the prover and the verifier can freely use perfect randomness, zero-knowledge proofs/arguments exist for all languages in NP under the minimum assumption of the existence of one-way functions [GMW91, Nao91, HILL99].

A natural question is whether zero-knowledge proofs/arguments can exist for non-trivial languages when randomness is only available in a restricted form. Previous work has investigated this question for several restricted uses of randomness. The study of cryptography with imperfect randomness [DOPS04] investigated this question in settings where only imperfect high-entropy randomness is available. The study of cryptography with tamperable randomness [ACM$^+$17] investigated this question in the presence of tampering attacks on randomness. (Strong impossibility results are known in both cases [DOPS04, ACM$^+$17].)

The study of *resettable security* [CGGM00] investigated this question in settings where zero-knowledge proofs/arguments are repeatedly executed with the same randomness. Zero-knowledge proofs/arguments are called *resettable zero-knowledge* [CGGM00] if they remain zero-knowledge when the prover randomness is reused. They are called *resettably sound* [MR01, BGGL01] if they remain sound when the verifier randomness is reused. They are called *simultaneously resettable zero-knowledge* [BGGL01, DGS09] if they are both resettable zero-knowledge and resettably sound. Resettable security is not only of theoretical interest but also of practical interest because generating perfect randomness often requires expensive operations (and is sometimes even impossible).

The resettable security of zero-knowledge proofs has been extensively studied, with strong positive results (e.g., [CGGM00, BGGL01, DL07, DGS09, PTW09, BOV12, COSV12, COPV13, COP$^+$14, BP15, OSV15, CPS16, COV17, BKS21]).[2] Of particular note, even simultaneously resettable zero-knowledge arguments are known to exist for all languages in NP under the assumption of the existence of one-way functions [COPV13]. The main technical point of these positive results is to use perfect randomness as the keys of pseudorandom functions (PRFs) and use pseudorandomness everywhere else. In particular, the pseudorandomness is generated by applying the PRFs to the communication transcript; as a result, even if the same randomness is reused, each execution is run with computationally independent randomness whenever adversarial parties send different messages.

However, only a few positive results are known for *resettable statistical zero-knowledge* [GOVW12]. Resettable statistical zero-knowledge is a natural strengthening of resettable (computational) zero-knowledge, and it guarantees statistical zero-knowledge even when the prover randomness is reused. Resettable statistical zero-knowledge is theoretically well-motivated because it helps us understand which notion of zero-knowledge can be achieved with limited uses of randomness. Practically, an advantage of resettable statistical zero-knowledge (over its computational counterpart) is *everlasting security* [MU10]; i.e., no security is compromised even if the underlying hardness assumptions are broken after protocol executions.[3] It is known that resettable statistical zero-knowledge proofs exist for all languages that admit hash proof systems [GOVW12].[4]

Compared with its computational counterpart, resettable statistical zero-knowledge is hard to realize since techniques developed in the computational setting are not helpful in the statistical setting. For example, even if each execution is run with computationally independent randomness, this does not seem sufficient to realize resettable statistical zero-knowledge.

1.1 Our Results

We study the problem of constructing resettable statistical zero-knowledge arguments for NP (i.e., for all languages in NP).

In the positive direction, we observe that given a *witness encryption scheme* for NP [GGSW13], a resettable statistical zero-knowledge argument for NP can be obtained by straightforwardly extending existing tech-

[1] If zero-knowledge is only required to hold against honest verifiers or bounded-auxiliary-input verifiers, deterministic-prover zero-knowledge arguments for non-trivial languages are known to exist [FNV17, DL20, BC20].

[2] We focus on those that study resettable security in the plain model, i.e., without relying on any trusted setup (such as common reference strings).

[3] In order to break computational soundness, cheating provers need to break the underlying hardness assumptions during protocol executions.

[4] When the honest prover strategy is allowed to be computationally unbounded, positive results are also known for, e.g., SZK.

niques [GOVW12].[5]

Theorem 1. *Assume the existence of one-way functions. Then, if there exists a witness encryption scheme for an* NP *language* $\mathbf{L}$*, there also exists a resettable statistical zero-knowledge argument for* $\mathbf{L}$*.*[6]

Witness encryption schemes are a generalization of public-key encryption schemes, and a message encrypted with an NP statement can be decrypted by using any corresponding witness. (The encrypted message is hidden if the statement is false.) Witness encryption schemes for NP can be obtained from *indistinguishability obfuscations* [GGH+13], which are known to exist under certain sets of well-founded assumptions [JLS21, JLS22]. Also, recent work has given witness encryption schemes for NP under new lattice assumptions (which are not known to imply indistinguishability obfuscations) [Tsa22, VWW22].

A natural question for our positive result is whether the use of witness encryption schemes is essential. It is known that both resettable zero-knowledge arguments for NP and statistical zero-knowledge arguments for NP can be obtained from one-way functions [CPS16, HNO+09]. Therefore, at first sight, it seems reasonable to conjecture that resettable statistical zero-knowledge arguments for NP can also be constructed from one-way functions.

We provide a strong negative result in this direction. As our main technical contribution, we show that even resettable statistical witness-indistinguishable arguments for NP require witness encryption schemes for NP.

Theorem 2. *Assume the existence of one-way functions. Then, if there exists a resettable statistical witness-indistinguishable argument for all languages in* NP*, there also exists a witness encryption scheme for all languages in* NP.

Together with Theorem 1, this result implies an equivalence of resettable statistical zero-knowledge arguments for NP and witness encryption schemes for NP. It also implies that, unless witness encryption schemes for NP are obtained from one-way functions, resettable statistical zero-knowledge arguments for NP require stronger primitives than resettable (computational) zero-knowledge arguments for NP do. (This is in contrast to the case of *concurrent zero-knowledge* [DNS98], a notion closely related to resettable zero-knowledge.[7] In the case of concurrent zero-knowledge, positive results for *statistical concurrent zero-knowledge* [GMOS07] (and even *statistical concurrent non-malleable zero-knowledge* [OOR+14]) are known for NP under the existence of one-way functions [GMOS07, Kiy20].)

Finally, to find a way to circumvent the above negative result, we consider resettable statistical security in a "fixed-input" setting. In the standard definition of resettable computational/statistical zero-knowledge [CGGM00, GOVW12], the prover randomness is reused to generate multiple proofs for multiple statements and witnesses. In the fixed-input setting, the prover randomness is reused to generate multiple proofs for a single statement and witness. (Verifier messages may be different in these multiple proofs.) Our proof of the above negative result does not hold in the fixed-input setting.

We show that *resettable statistically hiding commitment schemes* cannot exist even in the fixed-input setting.

Theorem 3 (informal)**.** *When multiple commitments are generated with the same committer randomness and committer input but different receiver messages, no computationally binding commitment scheme can be statistically hiding.*

Thus, even in the fixed-input setting, we cannot hope to use the naive approach of obtaining a resettable statistical zero-knowledge argument for NP from a resettable statistically hiding commitment scheme. Whether resettable statistical zero-knowledge arguments for NP can be constructed in the fixed-input setting from a weaker primitive than witness encryption schemes for NP is left as an interesting open problem.

2 Overview of Our Techniques

Section 2.1 explains the positive result: a resettable statistical zero-knowledge argument for an NP language $\mathbf{L}$ from a witness encryption scheme for $\mathbf{L}$. Section 2.2 and Section 2.3 explain the negative results: the im-

[5] This observation is likely to be already well-known among experts in the area.

[6] While a prior positive result [GOVW12] gives a resettable statistical zero-knowledge *proof* for a subclass of NP, this result gives a resettable statistical zero-knowledge *argument* for NP.

[7] In concurrent zero-knowledge, multiple proofs are generated using independent randomness in each execution. In resettable zero-knowledge, multiple proofs are generated using the same randomness.

possibility of resettable statistically hiding commitment schemes and the negative result on resettable statistical witness-indistinguishable arguments for NP.

2.1 Resettable Statistical Zero-Knowledge from Witness Encryption

As mentioned in Section 1, we construct a resettable statistical zero-knowledge argument for an NP language $\mathbf{L}$ from a witness encryption scheme for $\mathbf{L}$ by naturally extending existing techniques [GOVW12]. Although this construction does not involve new techniques, we provide a brief overview because it provides some intuition regarding our negative result on resettable statistical witness-indistinguishable arguments for NP (Section 2.3).

Preliminaries. First, we explain the existing techniques we rely on, namely those developed by Garg, Ostrovsky, Visconti, and Wadia (GOVW) [GOVW12]. Specifically, we recall their resettable statistical zero-knowledge proof for all languages that admit hash proof systems.

At a high level, the construction by GOVW [GOVW12] is similar to the classical interactive proof for Graph Non-Isomorphism by Goldreich, Micali, and Wigderson [GMW91]. In particular, it starts with the verifier committing to a random string m in a way that the following hold.

1. When the statement is true, any honest prover can efficiently obtain m.
2. When the statement is false, the committed string m is statistically hidden.

The prover is expected to reply with m, and the verifier accepts if and only if the reply is indeed equal to m. The first property above guarantees completeness, and the second property guarantees soundness. In addition, in the construction by GOVW [GOVW12], the commitment by the verifier is extractable by a rewinding simulator even against resetting committers,[8] and this extractability guarantees resettable statistical zero-knowledge.[9] More details are given below.

In the construction by GOVW [GOVW12], several *instance-dependent primitives* [IOS97] are used as building blocks. In instance-dependent primitives, each party receives an instance x of a language $\mathbf{L}$ as additional common input, and the security depends on whether the instance belongs to the language. The construction by GOVW [GOVW12] uses the following instance-dependent primitives.

- **An instance-dependent non-interactive extractable commitment scheme $\mathsf{Com}_{\mathbf{L}}$.** When $x \in \mathbf{L}$, the committed value can be efficiently extracted using any corresponding witness. When $x \notin \mathbf{L}$, the committed value is statistically hidden.
- **An instance-dependent resettably extractable commitment scheme $\mathsf{RECom}_{\mathbf{L}}$.**[10] When $x \in \mathbf{L}$, the committed value can be extracted from any resetting committer using rewinding techniques. When $x \notin \mathbf{L}$, the committed value is statistically hidden.
- **An instance-dependent resettably sound statistical witness-indistinguishable argument $\mathsf{rs\text{-}SWI}_{\mathbf{L}}$ for NP.** When $x \in \mathbf{L}$, resettable (computational) soundness holds. When $x \notin \mathbf{L}$, statistical witness indistinguishability holds. (Here, x is the instance given as additional common input, not the NP statement being proven.)

As observed by GOVW [GOVW12], these instance-dependent primitives exist for all languages that admit hash proof systems.

Given these building blocks, the construction by GOVW [GOVW12] can be described as follows. Let $x \in \mathbf{L}$ be the statement to be proven, and suppose it is also used as additional common input in each instance-dependent primitive.

1. $V \to P$: The verifier commits to a random string m using $\mathsf{Com}_{\mathbf{L}}$.

[8] In this overview, a *resetting adversary* is informally defined as an adversarial party that can force honest parties to reuse the same randomness in multiple executions.

[9] Following GOVW [GOVW12], we consider resettable statistical zero-knowledge in the model where cheating verifiers run in polynomial time and distinguishers run in unbounded time.

[10] GOVW [GOVW12] does not explicitly define this primitive, and implicitly obtains it by combining $\mathsf{Com}_{\mathbf{L}}$, a pseudorandom function, and a sophisticated rewinding technique developed in the context of concurrent zero-knowledge [PRS02].

2. $V \to P$: The verifier commits to m using $\mathsf{ REC om}_{\mathbf{L}}$.

3. $V \to P$: Using $\mathsf{rs\text{-}SWI}_{\mathbf{L}}$, the verifier proves that the above two steps were done correctly.

4. $P \to V$: The prover extracts m from the $\mathsf{Com}_{\mathbf{L}}$ commitment and sends m to the verifier.

The completeness, soundness, and resettable statistical zero-knowledge follow naturally from the security of the underlying instance-dependent primitives. The construction by GOVW [GOVW12] does not work for all NP since the above instance-dependent primitives are not known to exist for all NP.

Our construction. We use a witness encryption scheme for NP and a one-way function to convert the construction by GOVW [GOVW12] into a one for NP. The key point is that a witness encryption scheme can be used in place of the instance-dependent non-interactive extractable commitment scheme. Indeed, even in witness encryption schemes, encrypted values can be extracted using any corresponding witness when $x \in \mathbf{L}$, and encrypted values are hidden when $x \notin \mathbf{L}$. The only difference is that when $x \notin \mathbf{L}$, witness encryption schemes are computational hiding rather than statistical hiding. Fortunately, unlike GOVW [GOVW12] (whose goal was to obtain a resettable statistical zero-knowledge proof for a subclass of NP), we aim to construct a resettable statistical zero-knowledge argument for NP. In the security analysis of resettable statistical zero-knowledge arguments, the prover (which acts as a receiver in the witness encryption scheme) is computationally bounded. Thus, computational hiding is sufficient for our purpose.[11] In the same way, we can replace the other two instance-dependent primitives with constructions that (i) are instance-dependent w.r.t. all languages in NP (or not instance dependent at all) and (ii) can be obtained from one-way functions and witness encryption schemes for NP.[12] This way, we can obtain a construction that works for all languages in NP.

2.2 Impossibility of Resettable Statistically Hiding Commitment

Next, we explain the impossibility of resettable statistically hiding commitment schemes. This impossibility is a good warm-up for our negative result on resettable statistical witness-indistinguishable arguments for NP (Section 2.3).

We show that resettable statistical hiding and computational binding cannot be achieved simultaneously. Recall that statistical hiding implies that a random commitment to 1 can be "explained" as a commitment to 0. More formally, when we use notation $\tau_b(\mathsf{rnd}_C, \mathsf{rnd}_R)$ to denote the commitment generated with committer input b, committer randomness rnd_C, and receiver randomness rnd_R, we have that a random 1-commitment $\tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_R)$ is equal to a 0-commitment $\tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_R)$ for certain $\mathsf{rnd}_C^{(0)}$ with high probability, i.e.,

$$\Pr_{\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_R}\left[\begin{array}{l}\exists \mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C} \text{ s.t.} \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_R) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_R)\end{array}\right] \geq 1 - \mathsf{negl}(\lambda).$$

(Here, λ is the security parameter and ℓ_C is the length of committer randomness.) Resettable statistical hiding requires the above to hold even when the same committer randomness is used for multiple (say, t) commitments, i.e.,

$$\Pr_{\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_{R,1}, \ldots, \mathsf{rnd}_{R,t}}\left[\begin{array}{l}\exists \mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C} \text{ s.t. } \forall i \in \{1, \ldots, t\} : \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_{R,i}) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_{R,i})\end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \quad (1)$$

We show that the left-hand side of (1) is negligible when computational binding holds. Fix $\mathsf{rnd}_C^{(1)}$ arbitrarily in

[11] Witness encryption schemes were previously used in similar ways in the context of deterministic-prover zero-knowledge arguments/proofs [FNV17, DL20, BC20].

[12] For technical reasons, when the commitments in Steps 1 and 2 are computationally hiding, the consistency proof in Step 3 must guarantee zero-knowledge. Fortunately, a resettably sound zero-knowledge argument for NP can be obtained from one-way functions [CPS16], and we use it in our construction.

the left-hand side of (1). We have

$$\begin{aligned}
&\Pr_{\mathsf{rnd}_{R,1},\ldots,\mathsf{rnd}_{R,t}}\left[\begin{array}{l}\exists \mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C} \text{ s.t. } \forall i \in \{1,\ldots,t\}: \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_{R,i}) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_{R,i})\end{array}\right] \\
\le &\sum_{\mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C}} \Pr_{\mathsf{rnd}_{R,1},\ldots,\mathsf{rnd}_{R,t}}\left[\begin{array}{l}\forall i \in \{1,\ldots,t\}: \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_{R,i}) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_{R,i})\end{array}\right] \\
= &\sum_{\mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C}} \left(\Pr_{\mathsf{rnd}_R}\left[\tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_R) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_R)\right]\right)^t . \qquad (2)
\end{aligned}$$

(The second line follows from the union bound. The third line holds since each $\mathsf{rnd}_{R,i}$ is sampled independently.) Let us use computational binding to obtain an upper bound on (2). Since even a non-uniform cheating committer cannot break computational binding, we have the following for any $\mathsf{rnd}_C^{(0)}$ and $\mathsf{rnd}_C^{(1)}$.

$$\Pr_{\mathsf{rnd}_R}\left[\tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_R) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_R)\right] \le \mathsf{negl}(\lambda) \le \frac{1}{2}. \qquad (3)$$

(Indeed, if (3) does not hold, a cheating committer that commits to 0 using randomness $\mathsf{rnd}_C^{(0)}$ can open it to 1 using $\mathsf{rnd}_C^{(1)}$ with non-negligible probability.) It follows from (3) that (2) is at most $2^{-\lambda}$ when t is sufficiently large (concretely, $t \ge \ell_C + \lambda$). Since (2) holds for any $\mathsf{rnd}_C^{(1)}$, the left-hand side of (1) is also at most $2^{-\lambda}$. Therefore, we conclude that resettable statistical hiding does not hold when computational binding holds.

2.3 Witness Encryption from Resettable Statistical Witness Indistinguishability

Finally, we explain that resettable statistical witness-indistinguishable arguments for NP imply witness encryption schemes for NP.

We begin by noting that the techniques described in Section 2.2 do not work. By arguing as in Section 2.2, we can show the incompatibility of resettable statistical witness indistinguishability and computational binding. However, since interactive arguments are not required to be binding, the incompatibility does not lead to a contradiction.

Given this difficulty, we use more general resetting attacks than in Section 2.2. In particular, the prover randomness is reused to prove multiple statements. (Note that in Section 2.2, the committer randomness is only reused to generate multiple commitments for the same committer input $b \in \{0,1\}$.)

Overall approach. As a starting point, we observe that in our positive result (Section 2.1), the transcript between the prover and the verifier does not depend on the witness that the prover uses. That is, when the witness used by the prover is different but the randomness is the same, the same transcript is generated. (This is because the witness is only used during the extraction of m.) Roughly speaking, we show that (i) any interactive argument for NP satisfying such a "witness-independence" property can be used to construct a witness encryption scheme for NP, and (ii) resettable statistical witness-indistinguishable arguments for NP must satisfy such a property. More details are explained below.

Step 1: witness encryption from "witness-independent" argument. We show this step by using *predictable arguments* [FNV17]. Predictable arguments are private-coin interactive arguments such that the verifier can predict prover messages by using its private randomness. (An example is the interactive proof for Graph Non-Isomorphism by Goldreich, Micali, and Wigderson [GMW91].[13]) It is known that a witness encryption scheme for NP can be obtained from a predictable argument for NP [FNV17].[14] Therefore, we show that a predictable argument for NP can be obtained from any interactive argument for NP that satisfies a certain "witness-

[13]There, on input two graphs (G_0, G_1), the verifier sends a random isomorphic copy of G_b for a random $b \in \{0,1\}$ and checks whether the prover replies with b.

[14]This implication is shown by using that (i) given a true statement, the secret value predicted by the verifier can be efficiently obtained using any corresponding witness (this is because of correctness), and (ii) given a false statement, the secret value predicted by the verifier is computationally hidden (this is because of soundness).

independence" property. Toward this end, consider proving a statement $x \in \mathbf{L}$ by using an interactive argument (P, V) as follows.[15]

1. The verifier generates a "trapdoor statement" $x' \in \mathbf{L}'$ and a corresponding witness w' for a certain NP language $\mathbf{L}'$. Concretely, the verifier evaluates a pseudorandom generator $x' := \mathsf{PRG}(w')$ using a random seed w' and views x' as an instance of $\mathbf{L}' := \{x' \mid \exists w' \text{ s.t. } \mathsf{PRG}(w') = x'\}$.

2. The verifier executes (P, V) in its own head with statement $x \in \mathbf{L} \vee x' \in \mathbf{L}'$ and witness w'. (That is, the verifier internally emulates both the prover and the verifier of (P, V) using x, x', and w'.) Let the prover randomness and transcript of this execution be denoted by rnd_P and $(\beta_1, \alpha_1, \ldots, \beta_\rho, \alpha_\rho)$, respectively. (Each β_i is a verifier message, each α_i is a prover message, and ρ is the round complexity of (P, V).)

3. The verifier sends x' and rnd_P to the prover. Then, for each $i \in \{1, \ldots, \rho\}$ in sequence, the verifier sends β_i to the prover and checks whether the prover replies with α_i. (The prover, given as input a witness w for $x \in \mathbf{L}$, is expected to obtain α_i by executing (P, V) with statement $x \in \mathbf{L} \vee x' \in \mathbf{L}'$, witness w, prover randomness rnd_P, and verifier messages $\beta_1, \ldots, \beta_i$.)

Note that the verifier only checks whether the prover's replies agree with the predicted values $\alpha_1, \ldots, \alpha_\rho$. This construction is a predictable argument if (P, V) satisfies the following conditions.

Condition 1 (witness-independence condition)**.** *Let $\tau_x(x', w, \mathsf{rnd}_P, \mathsf{rnd}_V)$ denote the transcript of (P, V) generated with statement $x \in \mathbf{L}' \vee x' \in \mathbf{L}'$, witness w, prover randomness rnd_P, and verifier randomness rnd_V. Then, the following holds for any statement $x \in \mathbf{L}$ and any corresponding witness w.*

$$\Pr_{(x', w'), \mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau_x(x', w, \mathsf{rnd}_P, \mathsf{rnd}_V) = \tau_x(x', w', \mathsf{rnd}_P, \mathsf{rnd}_V)\right] \geq 1 - \mathsf{negl}(\lambda),$$

where x' and w' are sampled as in the above construction.

Indeed, the completeness follows from the above condition since it guarantees that an honest prover can obtain each α_i using w. Also, it can be shown that the soundness follows from the soundness of (P, V). Thus, given any interactive argument for NP that satisfies the above witness-independence condition, we can obtain a predictable argument for NP, and therefore, a witness encryption scheme for NP as well.

Step 2: "witness independence" of resettable statistical witness indistinguishability. It remains to show that any resettable statistical witness-indistinguishable argument for NP satisfies the above witness-independence condition. Considering the contrapositive, we show that an interactive argument (P, V) for NP is not resettable statistical witness indistinguishable if it does not satisfy the witness-independence condition. By definition, when (P, V) does not satisfy the witness-independence condition, there exists a statement $x \in \mathbf{L}$ and a corresponding witness w such that the following holds.

$$\Pr_{(x', w'), \mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau_x(x', w, \mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau_x(x', w', \mathsf{rnd}_P, \mathsf{rnd}_V)\right] \geq \frac{1}{\mathsf{poly}(\lambda)}. \tag{4}$$

For simplicity, we assume something stronger in this technical overview. In particular, we assume that there exists a statement $x \in \mathbf{L}$ and a corresponding witness w such that for sufficiently large t, there exist t instances $x'_1, \ldots, x'_t \in \mathbf{L}'$ and corresponding witnesses $w'_1, \ldots, w'_t$ such that the following holds for all $i \in \{1, \ldots, t\}$.

$$\Pr_{\mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau_x(x'_i, w, \mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau_x(x'_i, w'_i, \mathsf{rnd}_P, \mathsf{rnd}_V)\right] = 1. \tag{5}$$

(That is, the probability in the left-hand side of (4) is 1 for t statements and witnesses.) Under this assumption, we show that (P, V) is not resettable statistical witness indistinguishable.[16] Recall that, like statistical hiding in Section 2.2, statistical witness indistinguishability requires that a proof generated with one witness $w^{(0)}$ can be "explained" as a proof generated with another witness $w^{(1)}$. Resettable statistical witness indistinguishability requires that the same holds even when the same randomness is used to prove multiple statements. Thus, if (P, V) is resettable statistical witness indistinguishable, it satisfies the following for any $x \in \mathbf{L}$ and $x'_1, \ldots, x'_t \in \mathbf{L}'$.

---

[15] Similar constructions were previously considered in the contexts of deterministic-prover zero-knowledge [BC20] and witness maps [CPW20].

[16] The general case can be handled with a little care.

Condition 2 (necessary condition of resettable statistical witness indistinguishability)**.** *Let* $(w_1^{(0)}, \ldots, w_t^{(0)})$ *and* $(w_1^{(1)}, \ldots, w_t^{(1)})$ *be any two* t*-tuples of witnesses such that each* $w_i^{(0)}$ *and* $w_i^{(1)}$ *are witnesses for* $x \in \mathbf{L} \vee x'_i \in \mathbf{L}'$*. Then, the following holds.*

$$\Pr_{\mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V}\left[\begin{array}{l}\exists \mathsf{rnd}_P^{(0)} \text{ s.t. } \forall i \in \{1, \ldots, t\} : \\ \tau_x(x'_i, w_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_x(x'_i, w_i^{(1)}, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V)\end{array}\right] \geq 1 - \mathsf{negl}(\lambda). \tag{6}$$

Let us consider the above condition for the instances x and $x'_1, \ldots, x'_t$ satisfying (5). As in Section 2.2, we show that the left-hand side of (6) is negligible. Toward this end, we start by carefully defining $w_1^{(0)}, \ldots, w_t^{(0)}$ and $w_1^{(1)}, \ldots, w_t^{(1)}$. Specifically, using witnesses w and $w'_1, \ldots, w'_t$ that satisfy (5), we define them as follows.

- $w_i^{(0)}$ is an arbitrary witness for $x \in \mathbf{L} \vee x'_i \in \mathbf{L}'$.
- $w_i^{(1)}$ is defined by randomly sampling $b_i \in \{0, 1\}$ and setting $w_i^{(1)} := w$ if $b_i = 0$ and $w_i^{(1)} := w'_i$ if $b_i = 1$.

The key point is that $w_i^{(0)}$ and $w_i^{(1)}$ are defined so that two transcripts generated with them disagree with high probability. Indeed, since the transcript generated with w and that generated with w'_i disagree as shown in (5), at least one of them disagrees with the transcript generated with $w_i^{(0)}$. Thus, for any $\mathsf{rnd}_P^{(0)}$, $\mathsf{rnd}_P^{(1)}$, and rnd_V, we have the following.

$$\Pr_{b_i}\left[\tau_x(x'_i, w_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_x(x'_i, w_i^{(1)}, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V)\right] \leq \frac{1}{2}. \tag{7}$$

Given (7), we can proceed as in Section 2.2. Specifically, the left-hand side of (6) satisfies the following when $w_1^{(0)}, \ldots, w_t^{(0)}$ and $w_1^{(1)}, \ldots, w_t^{(1)}$ are defined as above by sampling $b_1, \ldots, b_t \in \{0, 1\}$.

$$\begin{aligned}
&\Pr_{b_1, \ldots, b_t, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V}\left[\begin{array}{l}\exists \mathsf{rnd}_P^{(0)} \text{ s.t. } \forall i \in \{1, \ldots, t\} : \\ \tau_x(x'_i, w_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_x(x'_i, w_i^{(1)}, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V)\end{array}\right] \\
&\leq \sum_{\mathsf{rnd}_P^{(0)}} \Pr_{b_1, \ldots, b_t, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V}\left[\begin{array}{l}\forall i \in \{1, \ldots, t\} : \\ \tau_x(x'_i, w_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_x(x'_i, w_i^{(1)}, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V)\end{array}\right] \\
&\leq \sum_{\mathsf{rnd}_P^{(0)}} \frac{1}{2^t}.
\end{aligned} \tag{8}$$

(The third line holds since we have (7) for any $\mathsf{rnd}_P^{(0)}$, $\mathsf{rnd}_P^{(1)}$, and rnd_V.) From an average argument, we can fix $b_1, \ldots, b_t$ so that (8) holds when $w_1^{(1)}, \ldots, w_t^{(1)}$ are defined based on the fixed values. That is, we have

$$\Pr_{\mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V}\left[\begin{array}{l}\exists \mathsf{rnd}_P^{(0)} \text{ s.t. } \forall i \in \{1, \ldots, t\} : \\ \tau_x(x'_i, w_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_x(x'_i, w_i^{(1)}, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V)\end{array}\right] \leq \sum_{\mathsf{rnd}_P^{(0)}} \frac{1}{2^t}.$$

Thus, the left-hand side of (6) is negligible when t is sufficiently large. Thus, (P, V) is not resettable statistical witness indistinguishable when it does not satisfy the witness-independence condition. This concludes the technical overview.

Remark 1 (On the possibility of constructing witness encryption for $\mathbf{L}$ from resettable statistical witness-indistinguishable argument for $\mathbf{L}$). As shown above, in the proof of Theorem 2, we obtain a witness encryption scheme for $\mathbf{L}$ by using a resettable statistical witness-indistinguishable argument for a related NP language. This is sufficient to prove Theorem 2 since it only states that the existence of resettable statistical witness-indistinguishable arguments for NP implies the existence of witness encryption schemes for NP.

However, theoretically speaking, it is more desirable if we obtain a witness encryption scheme for $\mathbf{L}$ by using a resettable statistical witness-indistinguishable argument for $\mathbf{L}$. (The positive result in Theorem 1 is shown in such a form.)

Unfortunately, our techniques seem insufficient to prove this desirable version of the result. A problem occurs in Step 1, i.e., the transformation from a "witness-independent" argument (P, V) to a predictable argument. Recall that in the transformation, the verifier of the predictable argument obtains a transcript of (P, V) in its head to predict prover messages. The problem is that if (P, V) is used for $\mathbf{L}$ (the language for which the predictable argument is designed), the verifier cannot obtain a transcript of (P, V) because it does not know a witness for $\mathbf{L}$. In this paper, we avoid this problem by having the verifier prepare a "trapdoor statement" x' and use (P, V) for $x \in \mathbf{L} \vee x' \in \mathbf{L}'$. As a result, (P, V) is required to work for a slightly larger language than the predictable argument.

A potential way to circumvent the above problem is to start with resettable statistical zero-knowledge instead of resettable statistical witness indistinguishability. If (P, V) is a resettable statistical zero-knowledge argument, the verifier of the predictable argument can use the simulator of (P, V) to obtain a transcript. (This is the approach used by Bitansky and Choudhuri [BC20] to obtain a predictable argument for $\mathbf{L}$ from a deterministic-prover zero-knowledge argument for $\mathbf{L}$.) In this case, however, it seems that the prover can obtain the same transcript as the verifier only when the prover can determine the prover randomness used in the simulated transcript. (In the context of deterministic-prover zero-knowledge, this problem does not arise because prover randomness does not exist.) Although the prover randomness is indeed efficiently recoverable in the case of the construction shown in our positive result (Theorem 1), we do not know if the same property holds for all resettable statistical zero-knowledge arguments. Therefore, even if we start with resettable statistical zero-knowledge, it is not clear if we can show Theorem 2 in the above desirable form. $\diamondsuit$

3 Preliminaries

3.1 Notations and Conventions

We use λ to denote the security parameter. For any binary strings $a, b \in \{0,1\}^*$, we use $a \| b$ to denote their concatenation. For any $n \in \mathbb{N}$, we use $[n]$ to denote the set $\{1, \ldots, n\}$. We use poly to denote an unspecified polynomial, negl to denote an unspecified negligible function, and PPT as an abbreviation of "probabilistic polynomial-time." For any NP language $\mathbf{L}$, we use $\mathbf{R_L}$ to denote the corresponding witness relation. For any instance $x \in \mathbf{L}$, we use $\mathbf{R_L}(x)$ to denote the set of the witnesses for $x \in \mathbf{L}$. For any pair of probabilistic interactive Turing machines (P, V), we use $\mathsf{output}_V\left[P(x) \leftrightarrow V(y)\right]$ to denote the random variable representing the output of V in an interaction between $P(x)$ and $V(y)$. Similarly, we use $\mathsf{trans}\left[P(x) \leftrightarrow V(y)\right]$ to denote the random variable representing the transcript of an interaction between $P(x)$ and $V(y)$. For a set S, we denote by $s \leftarrow S$ the process of obtaining an element $s \in S$ by a uniform sampling from S. For any probabilistic algorithm Algo and an input x, we denote by $y \leftarrow \mathsf{Algo}(x)$ the process of obtaining an output y by running $\mathsf{Algo}(x)$ with uniform randomness. When we write $\mathsf{Algo}(x; \mathsf{rnd})$, it means that Algo is run with input x and random tape rnd.

The reader is assumed to have basic knowledge of cryptography, such as the definitions of statistical/computational indistinguishability, one-way functions, and pseudorandom generators. (For these definitions, see standard textbooks like [Gol01].) In this paper, every adversarial party is modeled as a non-uniform Turing machine, i.e., takes a non-uniform string (denoted by $z \in \{0,1\}^*$) as additional input. While we do not explicitly specify the length of non-uniform strings, the length is always assumed to be bounded by a polynomial in λ.

3.2 Witness Encryption

We recall the definition of witness encryption schemes [GGSW13].

Definition 1 (Witness Encryption)**.** *A* witness encryption scheme *for an* NP *language* $\boldsymbol{L}$ *consists of two polynomial-time algorithms.*

Encryption. *The algorithm* $\mathsf{Enc}(1^\lambda, x, m)$ *takes as input a security parameter* λ *(in unary), an unbounded-length string* x*, and a message* $m \in \{0,1\}$*, and outputs a ciphertext* ct.

Decryption. *The algorithm* $\mathsf{Dec}(\mathsf{ct}, w)$ *takes as input a ciphertext* ct *and an unbounded-length string* w*, and outputs a message* m *or the symbol* $\perp$.

These algorithms satisfy the following two conditions.

Correctness. *For any* $\lambda \in \mathbb{N}$, $m \in \{0,1\}$, $x \in \boldsymbol{L}$, *and* $w \in \boldsymbol{R_L}(x)$,

$$\Pr\left[\mathsf{Dec}(\mathsf{Enc}(1^\lambda, x, m), w) = m\right] = 1.$$

Soundness Security. *For any* PPT *adversary* $\mathcal{A}$, *there exists a negligible function* negl *such that for every* $\lambda \in \mathbb{N}$, $x \notin L$, *and* $z \in \{0,1\}^*$,

$$\left|\Pr\left[\mathcal{A}(\mathsf{Enc}(1^\lambda, x, 0), z) = 1\right] - \Pr\left[\mathcal{A}(\mathsf{Enc}(1^\lambda, x, 1), z) = 1\right]\right| \le \mathsf{negl}(\lambda).$$

A witness encryption scheme can be naturally extended to encrypt a string (rather than a bit) by encrypting each bit of the string independently.

3.3 Interactive Argument, Witness Indistinguishability, and Zero-Knowledge

We recall the definition of interactive arguments and their basic privacy notions (e.g., [Gol01]).

Definition 2 (Interactive Argument)**.** *For any* NP *language* $\boldsymbol{L}$, *a pair of* PPT *interactive Turing machines* (P, V) *is called an* interactive argument *for* $\boldsymbol{L}$ *if it satisfies the following.*

Completeness. *For any polynomial* poly, *there exists a negligible function* negl *such that for every* $\lambda \in \mathbb{N}$, $x \in \boldsymbol{L} \cap \{0,1\}^{\mathsf{poly}(\lambda)}$, *and* $w \in \boldsymbol{R_L}(x)$,

$$\Pr\left[\mathsf{output}_V\left[P(1^\lambda, x, w) \leftrightarrow V(1^\lambda, x)\right] = 1\right] \ge 1 - \mathsf{negl}(\lambda).$$

Soundness. *For every* PPT *interactive Turing machine* P^* *and polynomial* poly, *there exists a negligible function* negl *such that for every* $\lambda \in \mathbb{N}$, $x \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \boldsymbol{L}$, *and* $z \in \{0,1\}^*$,

$$\Pr\left[\mathsf{output}_V\left[P^*(1^\lambda, x, z) \leftrightarrow V(1^\lambda, x)\right] = 1\right] \le \mathsf{negl}(\lambda).$$

In the above, P *is called the* prover *and* V *is called the* verifier.

Definition 3 (Statistical Witness Indistinguishability)**.** *An interactive argument* (P, V) *for an* NP *language* $\boldsymbol{L}$ *is called* statistical witness indistinguishable *if for any* PPT *interactive Turing machine* V^*, *any sequence* $\{x_\lambda, w_\lambda^{(0)}, w_\lambda^{(1)}\}_{\lambda\in\mathbb{N}}$ *such that* $x_\lambda \in \boldsymbol{L} \cap \{0,1\}^{\mathsf{poly}(\lambda)}$ *and* $w_\lambda^{(0)}, w_\lambda^{(1)} \in \boldsymbol{R_L}(x)$, *and any sequence of non-uniform strings* $\{z_\lambda\}_{\lambda\in\mathbb{N}}$, *the following ensembles are statistically indistinguishable.*

- $\left\{\mathsf{output}_{V^*}\left[P(1^\lambda, x_\lambda, w_\lambda^{(0)}) \leftrightarrow V^*(1^\lambda, x_\lambda, z_\lambda)\right]\right\}_{\lambda\in\mathbb{N}}$.
- $\left\{\mathsf{output}_{V^*}\left[P(1^\lambda, x_\lambda, w_\lambda^{(1)}) \leftrightarrow V^*(1^\lambda, x_\lambda, z_\lambda)\right]\right\}_{\lambda\in\mathbb{N}}$.

Definition 4 (Zero-Knowledge)**.** *An interactive argument* (P, V) *for an* NP *language* $\boldsymbol{L}$ *is called* (computational) zero-knowledge *if for any* PPT *interactive Turing machine* V^*, *there exists a* PPT *Turing machine* $\mathcal{S}$ *such that for any sequence* $\{x_\lambda, w_\lambda\}_{\lambda\in\mathbb{N}}$ *such that* $x_\lambda \in \boldsymbol{L} \cap \{0,1\}^{\mathsf{poly}(\lambda)}$ *and* $w_\lambda \in \boldsymbol{R_L}(x)$ *and for any sequence of non-uniform strings* $\{z_\lambda\}_{\lambda\in\mathbb{N}}$, *the following ensembles are computationally indistinguishable.*

- $\{\mathsf{output}_{V^*}\left[P(1^\lambda, x_\lambda, w_\lambda) \leftrightarrow V^*(1^\lambda, x_\lambda, z_\lambda)\right]\}_{\lambda\in\mathbb{N}}$.
- $\{\mathcal{S}(1^\lambda, x_\lambda, z_\lambda)\}_{\lambda\in\mathbb{N}}$.

In the above, V^* *is called the* cheating verifier *and* $\mathcal{S}$ *is called the* simulator.

3.4 Resettable Security of Interactive Arguments

First, we recall the definition of resettable statistical zero-knowledge [GOVW12].

Definition 5 (Resettable Statistical Zero-Knowledge)**.** *An interactive argument (P, V) for an* NP *language* $\boldsymbol{L}$ *is called* resettable statistical zero-knowledge *if for every* PPT *interactive Turing machine V^*, there exists a* PPT *Turing machine $\mathcal{S}$ such that for any polynomials* poly_x *and* poly_t*, the following holds.*

> *Let $t_\lambda := \mathsf{poly}_t(\lambda)$. Fix any sequences of t_λ-tuple $\{(x_{\lambda,1}, \dots, x_{\lambda,t_\lambda})\}_{\lambda \in \mathbb{N}}$ and $\{(w_{\lambda,1}, \dots w_{\lambda,t_\lambda})\}_{\lambda \in \mathbb{N}}$ such that $x_{\lambda,i} \in \boldsymbol{L} \cap \{0,1\}^{\mathsf{poly}_x(\lambda)}$ and $w_{\lambda,i} \in \boldsymbol{R_L}(x_{\lambda,i})$ for every $i \in [t_\lambda]$. Then, for any sequence of non-uniform strings $\{z_\lambda\}_{\lambda \in \mathbb{N}}$, the following two ensembles, denoted by $\{D_\lambda^0\}_{\lambda \in \mathbb{N}}$ and $\{D_\lambda^1\}_{\lambda \in \mathbb{N}}$, are statistically indistinguishable.*
>
> **Distribution D_λ^0.**
>
> 1. *Randomly select t_λ random tapes $\mathsf{rnd}_1, \dots, \mathsf{rnd}_{t_\lambda}$ for the prover P, resulting in deterministic strategies $\{P^{(i,j)}\}_{i,j \in [t_\lambda]}$ that are defined by $P^{(i,j)}(\alpha) = P(1^\lambda, x_{\lambda,i}, w_{\lambda,i}, \alpha; \mathsf{rnd}_j)$ for every $i, j \in [t_\lambda]$.*[17] *Each $P^{(i,j)}$ is called an* incarnation *of P.*
> 2. *On input 1^λ, $x_{\lambda,1}, \dots, x_{\lambda,t_\lambda}$, and z_λ, machine V^* initiates $\mathsf{poly}(\lambda)$-many (arbitrarily interleaved) interactions with the $P^{(i,j)}$'s, where V^* is allowed to send arbitrary messages to each of $P^{(i,j)}$ and obtain the responses of $P^{(i,j)}$ to such messages. Once V^* decides it is done interacting with the $P^{(i,j)}$'s, it produces an output based on its view of these interactions.*
> 3. *The output of the distribution is the final output of V^*.*
>
> **Distribution D_λ^1.** *The output of the distribution is the output of $\mathcal{S}(1^\lambda, x_{\lambda,1}, \dots, x_{\lambda,t_\lambda}, z_\lambda)$.*

In the above, V^* is called the cheating verifier *and $\mathcal{S}$ is called the* simulator.

Next, we define resettable statistical witness indistinguishability. The following definition is based on the definition of resettable (computational) witness indistinguishability [CGGM00]. However, since we only give a negative result about resettable statistical witness indistinguishability (and therefore considering a weaker security definition makes our result stronger), the definition is weaker than the natural one; see Remark 2 below for details.

Definition 6 (Resettable Statistical Witness Indistinguishability)**.** *An interactive argument (P, V) for an* NP *language* $\boldsymbol{L}$ *is called* resettable statistical witness indistinguishable *if for every* PPT *interactive Turing machine V^* and any polynomials* $\mathsf{poly}_x, \mathsf{poly}_t$*, the following holds.*

> *Let $t_\lambda := \mathsf{poly}_t(\lambda)$. Fix any sequence of t_λ-tuple $\{(x_{\lambda,1}, \dots, x_{\lambda,t_\lambda})\}_{\lambda \in \mathbb{N}}$, $\{(w^0_{\lambda,1}, \dots w^0_{\lambda,t_\lambda})\}_{\lambda \in \mathbb{N}}$, and $\{(w^1_{\lambda,1}, \dots w^1_{\lambda,t_\lambda})\}_{\lambda \in \mathbb{N}}$ such that (i) $x_{\lambda,i} \in \boldsymbol{L} \cap \{0,1\}^{\mathsf{poly}_x(\lambda)}$ and $w^0_{\lambda,i}, w^1_{\lambda,i} \in \boldsymbol{R_L}(x_{\lambda,i})$ for every $i \in [t_\lambda]$ and (ii) $x_{\lambda,i} \neq x_{\lambda,j}$ for every distinct $i, j \in [t_\lambda]$. Then, for any sequence of non-uniform strings $\{z_\lambda\}_{\lambda \in \mathbb{N}}$, the following two ensembles, denoted by $\{D_\lambda^0\}_{\lambda \in \mathbb{N}}$ and $\{D_\lambda^1\}_{\lambda \in \mathbb{N}}$, are statistically indistinguishable.*
>
> **Distribution D_λ^b ($b \in \{0,1\}$).**
>
> 1. *Randomly select a random tape rnd for the prover P, resulting in deterministic strategies $P^{(1)}, \dots P^{(t_\lambda)}$ that are defined by $P^{(i)}(\alpha) = P(1^\lambda, x_{\lambda,i}, w^b_{\lambda,i}, \alpha; \mathsf{rnd})$ for every $i \in [t_\lambda]$.*
> 2. *On input 1^λ, $x_{\lambda,1}, \dots, x_{\lambda,t_\lambda}$, and z_λ, machine V^* initiates t_λ sequential interactions with the $P^{(i)}$'s, where the i-th interactions is done with $P^{(i)}$.*
> 3. *The output of the distribution is the final output of V^*.*

Remark 2. Compared with the natural definition that we can think of, Definition 6 is weak since V^* is only allowed to interact with each incarnation once in sequence. (Additionally, as a minor restriction, the statements $x_{\lambda,1}, \dots, x_{\lambda,t_\lambda}$ are required to be distinct.) ♢

[17] $P(1^\lambda, x_{\lambda,i}, w_{\lambda,i}, \alpha; \mathsf{rnd}_j)$ denotes the message sent by P on input $(1^\lambda, x_{\lambda,i}, w_{\lambda,i})$ and random tape rnd_j after seeing the message-sequence α.

Finally, we recall the definition of resettable soundness [BGGL01].

Definition 7 (Resettably Sound Argument)**.** *A* resetting attack *of a cheating prover P^* on a* resettable verifier *V is defined by the following two-step random experiment, indexed by a security parameter λ.*

1. *Uniformly select and fix $t = \mathsf{poly}(\lambda)$ random tapes $\mathsf{rnd}_1, \dots, \mathsf{rnd}_t$ for V, resulting in deterministic strategies $V^{(j)}(x) = V_{x,\mathsf{rnd}_j}$ defined by $V_{x,\mathsf{rnd}_j}(\alpha) = V(1^\lambda, x, \alpha; \mathsf{rnd}_j)$, where $x \in \{0,1\}^\lambda$ and $j \in [t]$. Each $V^{(j)}(x)$ is called an* incarnation *of V.*

2. *On input 1^λ, machine P^* initiates $\mathsf{poly}(\lambda)$-many sequential interactions with the $V^{(j)}(x)$'s. The activity of P^* proceeds in rounds. In each round, P^* chooses $x \in \{0,1\}^\lambda$ and $j \in [t]$, thus defining $V^{(j)}(x)$, and conducts a complete session with it. Once P^* decides it is done interacting with the $V^{(j)}(x)$'s, it produces an output based on its view of these interactions.*

Let (P, V) be an interactive argument for an NP language $\boldsymbol{L}$. We say that (P, V) is resettably sound *if the following two conditions hold.*

Resettable-completeness. *Consider an arbitrary polynomial-size resetting attack,*[18] *and suppose that in some session, after selecting an incarnation $V^{(j)}(x)$, the attacker follows the (honest) strategy P.*[19] *Then, if $x \in \boldsymbol{L}$, the probability that $V^{(j)}(x)$ rejects is negligible.*

Resettable-soundness. *For every polynomial-size resetting attack, the probability that in some session the corresponding $V^{(j)}(x)$ has accepted and $x \notin \boldsymbol{L}$ is negligible.*

Remark 3. Definition 7 is known to be equivalent to the (seemingly stronger) "interleaving" version of the definition, where P^* is allowed to initiate $\mathsf{poly}(\lambda)$-many arbitrarily interleaved interactions with the $V^{(j)}(x)$'s [BGGL01]. $\diamondsuit$

If a resettably sound argument is zero-knowledge (resp. statistically witness indistinguishable), it is called a resettably sound zero-knowledge (resp. statistical witness-indistinguishable) argument. It is known that a constant-round resettably sound zero-knowledge argument exists for all languages in NP under the assumption of one-way functions [CPS16].

3.5 Predictable Argument

We recall the definition of predictable arguments [FNV17]. The following definition is based on that given in [BC20], which is weaker than the original definition [FNV17] in that the existence of witness extractors is not required.

Definition 8 (Predictable Argument)**.** *A ρ-round* predictable argument *for an NP language $\boldsymbol{L}$ is specified by a tuple of algorithms $\Pi = (\mathsf{Chal}, \mathsf{Resp})$ as follows.*

1. *The verifier V samples $(\boldsymbol{\beta}, \boldsymbol{\alpha}) \leftarrow \mathsf{Chal}(1^\lambda, x)$, where $\boldsymbol{\beta} := (\beta_1, \dots, \beta_\rho)$ and $\boldsymbol{\alpha} := (\alpha_1, \dots, \alpha_\rho)$.*
2. *For all $i \in [\rho]$ in increasing sequence, the verifier V and the prover P do the following.*
   (a) V sends β_i to P.
   (b) P sends $\tilde{\alpha}_i := \mathsf{Resp}(1^\lambda, x, w, \beta_1, \dots, \beta_i)$ to V.
3. *V outputs 1 if $\tilde{\alpha}_i = \alpha_i$ for all $i \in [\rho]$, and outputs 0 otherwise.*

The algorithms are required to satisfy the following two conditions.

Completeness. *There exists a negligible function negl such that for any $\lambda \in \mathbb{N}$, $x \in \boldsymbol{L}$, and $w \in \boldsymbol{R_L}(x)$,*

$$\Pr\left[\mathsf{output}_V\left[P(1^\lambda, x, w) \leftrightarrow V(1^\lambda, x)\right] = 1\right] \geq 1 - \mathsf{negl}(\lambda).$$

[18]Polynomial-size resetting attacks are those such that the cheating provers take polynomial-length non-uniform inputs and run in polynomial time.

[19]To consider honest prover strategies that are implementable in probabilistic polynomial time, we need to supply P with an adequate NP witness. Thus, we consider a resetting attack that for every selected $x \in \mathbf{L}$ also provides P with $w \in \mathbf{R}_\mathbf{L}(x)$. In this case, we require that when $V^{(j)}(x)$ interacts with $P(x, w)$, it rejects with negligible probability.

Soundness. *For any* PPT *cheating prover* P^* *and any polynomial* poly*, there exists a negligible function* negl *such that for every* $\lambda \in \mathbb{N}$*,* $x \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \boldsymbol{L}$*, and* $z \in \{0,1\}^*$*,*

$$\Pr\left[\mathsf{output}_V\left[P^*(1^\lambda, x, z) \leftrightarrow V(1^\lambda, x)\right] = 1\right] \leq \mathsf{negl}(\lambda).$$

It is known that predictable arguments are equivalent to witness encryption schemes [FNV17].[20]

Theorem 4 ([FNV17])**.** *Let* $\boldsymbol{L}$ *be any* NP *language. There exists a predictable argument for* $\boldsymbol{L}$ *if and only if there exists a witness encryption scheme for* $\boldsymbol{L}$*.*

3.6 Commitment Scheme and Resettable Statistical Hiding

First, we recall the definition of statistically hiding (bit-)commitment schemes (e.g., [Gol01]).[21]

Definition 9 (Statistically Hiding Commitment)**.** *A pair of* PPT *interactive Turing machines* (C, R) *is called a* statistically hiding (bit-)commitment scheme *if it satisfies the following.*

Computational Binding. *Let* ℓ_R *denote the length of the random tape for* R*. Then, for every pair of* PPT *interactive Turing machines* (C_0^*, C_1^*)*, there exists a negligible function* negl *such that for every* $\lambda \in \mathbb{N}$ *and* $z \in \{0,1\}^*$*,*

$$\Pr\left[\tau = \tau_0 \wedge \tau = \tau_1 \;\middle|\; \begin{array}{l} \mathsf{rnd}_R \leftarrow \{0,1\}^{\ell_R(\lambda)} \\ \tau \leftarrow \mathsf{trans}\left[C_0^*(z) \leftrightarrow R(1^\lambda; \mathsf{rnd}_R)\right] \\ (\mathsf{rnd}_0, \mathsf{rnd}_1) \leftarrow C_1^*(\tau, z) \\ \tau_0 := \mathsf{trans}\left[C(1^\lambda, 0; \mathsf{rnd}_0) \leftrightarrow R(1^\lambda; \mathsf{rnd}_R)\right] \\ \tau_1 := \mathsf{trans}\left[C(1^\lambda, 1; \mathsf{rnd}_1) \leftrightarrow R(1^\lambda; \mathsf{rnd}_R)\right] \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Statistical Hiding. *For every* PPT *interactive Turing machine* R^* *and a sequence of non-uniform strings* $\{z_\lambda\}_{\lambda \in \mathbb{N}}$*, the following ensembles are statistically indistinguishable.*

- $\left\{\mathsf{output}_{R^*}\left[C(1^\lambda, 0) \leftrightarrow R^*(z_\lambda)\right]\right\}_{\lambda \in \mathbb{N}}$.
- $\left\{\mathsf{output}_{R^*}\left[C(1^\lambda, 1) \leftrightarrow R^*(z_\lambda)\right]\right\}_{\lambda \in \mathbb{N}}$.

In the above, C *is called the* committer *and* R *is called the* receiver.

Next, we define resettable statistical hiding. The definition is similar to the definition of resettable statistical witness indistinguishability (Definition 6). Since we only give a negative result about resettable statistical hiding, the definition is weaker than the natural one; see Remark 4 below for details.

Definition 10 (Resettable Statistical Hiding Commitment)**.** *A commitment scheme* (C, R) *is called* resettable statistical hiding *if for every* PPT *interactive Turing machine* R^* *and every sequence of non-uniform strings* $\{z_\lambda\}_{\lambda \in \mathbb{N}}$*, the following two ensembles, denoted by* $\{D_\lambda^0\}_{\lambda \in \mathbb{N}}$ *and* $\{D_\lambda^1\}_{\lambda \in \mathbb{N}}$*, are statistically indistinguishable.*

Distribution D_λ^b **(**$b \in \{0,1\}$**).**

1. *Randomly select a random tape* rnd *for the committer* C*, resulting in deterministic strategy* C_b *that is defined by* $C_b(\alpha) = C(1^\lambda, b, \alpha; \mathsf{rnd})$.
2. *On input* 1^λ *and* z_λ*, machine* R^* *initiates* $\mathsf{poly}(\lambda)$*-many sequential interactions with* C_b*. Once* R^* *decides it is done interacting with* C_b*, it produces an output based on its view of these interactions.*
3. *The output of the distribution is the output of* R^*.

Remark 4. Definition 10 is weak in that R^* is only allowed to interact with a single incarnation of C. $\Diamond$

[20] In [FNV17], the equivalence is shown for stronger versions of predictable arguments and witness encryption schemes (*predictable arguments of knowledge* and *extractable witness encryption schemes*, respectively). However, as mentioned in [FNV17], the equivalence also holds for predictable arguments and witness encryption schemes.

[21] For notational simplicity, we assume that the reveal phase proceeds as follows: (i) the committer reveals the committed value and the random tape that it used in the commit phase; (ii) the receiver checks whether the revealed committed value and random tape reproduce the transcript of the commit phase.

3.7 Instance-Dependent Primitives

We recall the definitions of several instance-dependent primitives.

3.7.1 Instance-dependent non-interactive commitment.

First, we recall the definition of instance-dependent non-interactive commitment schemes. The definition below is the version given in [GOVW12].

Definition 11 (Instance-dependent non-interactive commitment)**.** *A* PPT *Turing machines* Com *is called an* instance-dependent (perfectly binding statistically hiding) non-interactive commitment scheme *with respect to a language* $\boldsymbol{L}$ *if it satisfies the following.*

Perfect Binding. *For every* $\lambda \in \mathbb{N}$*,* $x \in \boldsymbol{L}$*,* $m_0, m_1 \in \{0,1\}^\lambda$*, and* $\mathsf{rnd}_0, \mathsf{rnd}_1 \in \{0,1\}^{\ell(\lambda)}$ *(where* ℓ *is the length of the random tape of* Com*), if* $m_0 \neq m_1$*, then* $\mathsf{Com}(1^\lambda, x, m_0; \mathsf{rnd}_0) \neq \mathsf{Com}(1^\lambda, x, m_1; \mathsf{rnd}_1)$*.*

Statistical Hiding. *For every* $\{x_\lambda\}_{\lambda\in\mathbb{N}}$ *and* $\{m_\lambda^{(0)}, m_\lambda^{(1)}\}_{\lambda\in\mathbb{N}}$ *such that* $x_\lambda \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \boldsymbol{L}$ *and* $m_\lambda^{(0)}, m_\lambda^{(1)} \in \{0,1\}^\lambda$*, the following two ensembles are statistically indistinguishable.*

$$\left\{\mathsf{Com}(1^\lambda, x_\lambda, m_\lambda^{(0)})\right\}_{\lambda\in\mathbb{N}} \qquad \text{and} \qquad \left\{\mathsf{Com}(1^\lambda, x_\lambda, m_\lambda^{(1)})\right\}_{\lambda\in\mathbb{N}}.$$

Additionally, Com *is called* (efficiently) extractable *if it satisfies the following.*

Extractability. *There exists a polynomial-time Turing machine* E *such that for every* $\lambda \in \mathbb{N}$*,* $x \in \boldsymbol{L}$*,* $w \in \boldsymbol{R_L}(x)$*, and* $m \in \{0,1\}^\lambda$*,*

$$\Pr\left[\tilde{m} = m \;\middle|\; c \leftarrow \mathsf{Com}(1^\lambda, x, m);\ \tilde{m} := E(c, w)\ \right] = 1.$$

In the above, E *is called the* extractor.

3.7.2 Instance-dependent resettably sound statistical witness-indistinguishable argument.

Next, we recall the definition of instance-dependent resettably sound statistical witness-indistinguishable arguments. (The following is the version given in [GOVW12].) *Instance-dependent resettably sound statistical witness-indistinguishable arguments* for an NP language $\mathbf{L}$ are defined w.r.t. another NP language $\mathbf{L}'$. The differences from (ordinary) resettably sound statistical witness-indistinguishable arguments are the following.

- In addition to receiving an instance x of $\mathbf{L}$ as the statement to be proven, the prover P and the verifier V take an instance x' of $\mathbf{L}'$ as an additional common input.
- The resettable-completeness and resettable-soundness are required to hold only when $x' \in \mathbf{L}'$.[22]
- The statistical witness indistinguishability is required to hold only when $x' \notin \mathbf{L}'$.[23]

If instance-dependent non-interactive commitment schemes exist w.r.t. a language $\mathbf{L}$, instance-dependent resettably sound statistical witness-indistinguishable arguments for NP also exist w.r.t. $\mathbf{L}$ [GOVW12].

4 Resettable Statistical Zero-Knowledge from Witness Encryption

This section shows that a resettable statistical zero-knowledge argument for NP can be constructed from a witness encryption scheme for NP.

Theorem 5 (restatement of Theorem 1)**.** *Assume the existence of one-way functions. Then, if there exists a witness encryption scheme for an* NP *language* $\boldsymbol{L}$*, there also exists a resettable statistical zero-knowledge argument for* $\boldsymbol{L}$*.*

[22] The definition of a resetting attack (Definition 7) is modified as follows. (1) A sequence $(x'_1, \ldots, x'_t)$ such that $x'_k \in \mathbf{L}'$ is fixed at the beginning of the experiment. (2) The incarnations of V are defined as $\{V^{(j,k)}(x)\}_{j,k\in[t]}$, where each $V^{(j,k)}(x) = V_{x,x'_k,\mathsf{rnd}_j}$ is defined by $V_{x,x'_k,\mathsf{rnd}_j}(\alpha) = V(x, x'_k, \alpha; \mathsf{rnd}_j)$. (3) When interacting with an incarnation of V, the cheating prover P^* chooses x, j, and k to define $V^{(j,k)}(x)$.

[23] That is, the requirement is that for any $x \in \mathbf{L}$ and $x' \notin \mathbf{L}$, a proof generated with common input (x, x') and private input $w_x^{(0)}$ is statistically indistinguishable from a proof generated with common input (x, x') and private input $w_x^{(1)}$.

4.1 Preliminary: Construction by Garg et al. [GOVW12]

As described in Section 2.1, our construction is based on a prior construction by Garg, Ostrovsky, Visconti, and Wadia (GOVW) [GOVW12]. We start by recalling their construction. Let $\mathbf{L}$ be any language such that an instance-dependent non-interactive commitment scheme exists w.r.t. $\mathbf{L}$. GOVW [GOVW12] gave a resettable statistical zero-knowledge proof for $\mathbf{L}$ using the following two building blocks, both of which are instance dependent w.r.t. $\mathbf{L}$.

- $\mathsf{Com}_{\mathbf{L}}$: An instance-dependent non-interactive extractable commitment scheme. It is (i) perfectly binding and extractable when a true instance $x \in \mathbf{L}$ is given to the committer and the receiver, and (ii) statistically hiding when a false instance $x \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \mathbf{L}$ is given to them (cf. Definition 11).

- $\mathsf{rs\text{-}SWI}_{\mathbf{L}} = (\mathsf{rs\text{-}SWI.P}_{\mathbf{L}}, \mathsf{rs\text{-}SWI.V}_{\mathbf{L}})$: An instance-dependent resettably sound statistical witness-indistinguishable argument for NP. It is (i) resettably sound when a true instance $x \in \mathbf{L}$ is given to the prover and the verifier as an additional common input, and (ii) statistically witness indistinguishable when a false instance $x \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \mathbf{L}$ is given to them (cf. Section 3.7).

The construction by GOVW [GOVW12] is given in Figure 1.[24]

4.2 Our Construction and Its Security

We are ready to prove Theorem 5.

Proof. Let $\mathbf{L}$ be any NP language. We obtain a resettable statistical zero-knowledge argument for $\mathbf{L}$ from the following building blocks.

- $\mathsf{WE} = (\mathsf{WE.Enc}, \mathsf{WE.Dec})$: A witness encryption scheme for $\mathbf{L}$.

- $\mathsf{rs\text{-}ZK} = (\mathsf{rs\text{-}ZK.P}, \mathsf{rs\text{-}ZK.V})$: A resettably sound zero-knowledge argument for NP (which can be obtained from one-way functions [CPS16]).

Our resettable statistical zero-knowledge argument for $\mathbf{L}$ is obtained by modifying the construction in Figure 1 as follows.

- $\mathsf{WE.Enc}$ is used instead of $\mathsf{Com}_{\mathbf{L}}$, and $\mathsf{WE.Dec}$ is used instead of its extractor.

- $\mathsf{rs\text{-}ZK}$ is used instead of $\mathsf{rs\text{-}SWI}_{\mathbf{L}}$.

The completeness and resettable statistical zero-knowledge can be verified by inspection. In particular, the resettable statistical zero-knowledge can be verified by observing that when $x \in \mathbf{L}$, WE and $\mathsf{rs\text{-}ZK}$ guarantee the same security as $\mathsf{Com}_{\mathbf{L}}$ and $\mathsf{rs\text{-}SWI}_{\mathbf{L}}$ against V^*, respectively. (The extractability of WE follows from its perfect correctness.)

Regarding the soundness, we prove it by using a hybrid argument. Assume for contradiction that there exists a PPT cheating prover P^* and a polynomial p such that for infinitely many $\lambda \in \mathbb{N}$, there exist $x \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \mathbf{L}$ and $z \in \{0,1\}^*$ such that $P^*(1^\lambda, x, z)$ makes an honest verifier $V(1^\lambda, x)$ output 1 with probability at least $1/p(\lambda)$. Fix any such P^*, λ, x, and z. Then, consider the following hybrid experiments $H_0, \ldots, H_{\kappa^2+2}$.

- Hybrid H_0 is the real soundness experiment, where $P^*(1^\lambda, x, z)$ interacts with an honest verifier $V(1^\lambda, x)$. From our assumption, V outputs 1 with probability at least $1/p(\lambda)$.

- Hybrid H_1 is identical with H_0 except that the execution of $\mathsf{rs\text{-}ZK}$ in Step 2 is simulated. From the zero-knowledge of $\mathsf{rs\text{-}ZK}$, there exists a negligible function negl_1 such that V outputs 1 with probability at least $1/p(\lambda) - \mathsf{negl}_1(\lambda)$.

- Hybrid H_2 is identical with H_1 except that in Step 1, the commitment c is computed by $c \leftarrow \mathsf{WE.Enc}(1^\lambda, x, 0^\lambda)$ rather than by $c \leftarrow \mathsf{WE.Enc}(1^\lambda, x, m)$. From the soundness security of WE, there exists a negligible function negl_2 such that V outputs 1 with probability at least $1/p(\lambda) - \mathsf{negl}_2(\lambda)$.

[24]The description of the construction differs slightly from that given in the technical overview (Section 2.1). Specifically, $\mathsf{RECom}_{\mathbf{L}}$ is instantiated in Steps 1(b) and 3 using $\mathsf{Com}_{\mathbf{L}}$, a pseudorandom function, and the so-called *PRS preamble* [PRS02].

The common input is a security parameter λ and an instance x of $\mathbf{L}$. The private input to the prover is a witness w for $x \in \mathbf{L}$. Let $\kappa = \omega(\log \lambda)$ be a parameter that is defined based on λ. (E.g., $\kappa := \lceil \log^2 \lambda \rceil$.)

1. **Determining Message:** V does the following.
   (a) Sample a uniformly random string $m \in \{0,1\}^\lambda$ and compute $c \leftarrow \mathsf{Com}_{\mathbf{L}}(1^\lambda, x, m)$.
   (b) For every $i \in [\kappa]$ and $j \in [\kappa]$, sample uniformly random strings $\sigma^0_{i,j}, \sigma^1_{i,j} \in \{0,1\}^\lambda$ such that $\sigma^0_{i,j} \oplus \sigma^1_{i,j} = m$ and compute $c^0_{i,j} \leftarrow \mathsf{Com}_{\mathbf{L}}(1^\lambda, x, \sigma^0_{i,j})$ and $c^1_{i,j} \leftarrow \mathsf{Com}_{\mathbf{L}}(1^\lambda, x, \sigma^1_{i,j})$.
   (c) Send $(c, \{c^0_{i,j}, c^1_{i,j}\}_{i,j\in[\kappa]})$ to P.
2. **Proof of Consistency:** V uses $\mathsf{rs\text{-}SWI}_{\mathbf{L}}$ to prove the following NP statement: There exist $\tilde{m}$, rnd, and $\{\tilde{\sigma}^0_{i,j}, \mathsf{rnd}^0_{i,j}, \tilde{\sigma}^1_{i,j}, \mathsf{rnd}^1_{i,j}\}_{i,j\in[\kappa]}$ that satisfy all of the following.
   (a) $\mathsf{Com}_{\mathbf{L}}(1^\lambda, x, \tilde{m}; \mathsf{rnd}) = c$.
   (b) $\mathsf{Com}_{\mathbf{L}}(1^\lambda, x, \tilde{\sigma}^b_{i,j}; \mathsf{rnd}^b_{i,j}) = c^b_{i,j}$ for every $i \in [\kappa]$, $j \in [\kappa]$, and $b \in \{0,1\}$.
   (c) $\tilde{\sigma}^0_{i,j} \oplus \tilde{\sigma}^1_{i,j} = \tilde{m}$ for every $i \in [\kappa]$ and $j \in [\kappa]$.
3. **Resettable PRS Phase:** P samples a random key $s \in \{0,1\}^\lambda$ of a pseudorandom function PRF and computes $\omega := \mathsf{PRF}(s, x \,\|\, \mathsf{msg})$, where $\mathsf{msg} := (c, \{c^0_{i,j}, c^1_{i,j}\}_{i,j\in[\kappa]})$. Next, P divides ω into κ blocks of κ-bit each, i.e., obtains $(\omega_1, \ldots, \omega_\kappa)$ such that $\omega = \omega_1 \| \cdots \| \omega_\kappa$ and $|\omega_i| = \kappa$ for every $i \in [\kappa]$. Then, for each $k \in [\kappa]$ in sequence, P and V do the following.
   (a) P sends ω_k to V.
   (b) V sends the opening of $c^{\omega_{k,1}}_{k,1}, \ldots, c^{\omega_{k,\kappa}}_{k,\kappa}$ to P, where $\omega_{k,j}$ is the j-th bit of ω_k for each $j \in [\kappa]$. (In other words, V sends $(\sigma^{\omega_{k,1}}_{k,1}, \mathsf{rnd}^{\omega_{k,1}}_{k,1}), \ldots, (\sigma^{\omega_{k,\kappa}}_{k,1}, \mathsf{rnd}^{\omega_{k,\kappa}}_{k,\kappa})$ to P, where $\mathsf{rnd}^{\omega_{k,j}}_{k,j}$ is the random tape used to compute $c^{\omega_{k,j}}_{k,j}$.)
   (c) The prover P aborts the protocol if the openings are invalid (i.e., $\exists j \in [\kappa]$ s.t. $\mathsf{Com}_{\mathbf{L}}(1^\lambda, x, \sigma^{\omega_{k,j}}_{i,j}; \mathsf{rnd}^{\omega_{k,j}}_{i,j}) \neq c^{\omega_{k,j}}_{i,j}$).
4. **Final Message:**
   (a) P runs the extractor of $\mathsf{Com}_{\mathbf{L}}$ on input (c, w) and sends the extracted message to V. (If the extractor aborts, P aborts the protocol.) The extracted message is denoted by m'.
   (b) V outputs 1 if and only if $m = m'$.

Figure 1: Construction by GOVW [GOVW12].

- Hybrid $H_{(i-1)\kappa+(j-1)+3}$ for each $i, j \in [\kappa]$ is identical with $H_{(i-1)\kappa+(j-1)+2}$ except that in Step 1, the random strings $(\sigma^0_{i,j}, \sigma^1_{i,j})$ are samples uniform randomly from $\{0,1\}^\lambda \times \{0,1\}^\lambda$ without the condition of $\sigma^0_{i,j} \oplus \sigma^1_{i,j} = m$. From the soundness security of WE, there exists a negligible function negl_3 such that V outputs 1 with probability at least $1/p(\lambda) - \mathsf{negl}_2(\lambda) - ((i-1)\kappa + (j-1) + 1) \cdot \mathsf{negl}_3(\lambda)$.

Thus, in H_{κ^2+2}, the honest verifier outputs 1 with non-negligible probability. This is a contradiction since in H_{κ^2+2}, we have $m = m'$ with probability at most $2^{-\lambda}$. (The cheating prover P^* obtains no information about m in H_{κ^2+2}.) □

5 Witness Encryption from Resettable Statistical Witness Indistinguishability

This section shows that resettable statistical witness-indistinguishable arguments for NP imply witness encryption schemes for NP.

Algorithm $\mathsf{Chal}(1^\lambda, x)$**:**

1. Sample an instance $\hat{x} \in \widehat{\mathbf{L}}$ and a corresponding witness $\hat{w}$ as follows. Sample a uniformly random string $s \in \{0,1\}^\lambda$ and compute $r := \mathsf{PRG}(s)$. Then, let $\hat{x} := (x, r)$ and $\hat{w} := s$.
2. Run $(\mathsf{rSWI.P}, \mathsf{rSWI.V})$ with prover input $(1^\lambda, \hat{x}, \hat{w})$ and verifier input $(1^\lambda, \hat{x})$ by emulating both the prover and the verifier internally. Let $\tau = (\beta_1, \alpha_1, \ldots, \beta_\rho, \alpha_\rho)$ be the resulting transcript. (Each β_i is sent by the verifier, and each α_i is sent by the prover.) Let rnd_P and rnd_V be the random tapes used by the prover and the verifier, respectively.
3. Output $(\boldsymbol{\beta}, \boldsymbol{\alpha})$, where $\boldsymbol{\beta} := ((r, \mathsf{rnd}_P, \beta_1), \beta_2, \ldots, \beta_\rho)$ and $\boldsymbol{\alpha} := (\alpha_1, \ldots, \alpha_\rho)$.

Algorithm $\mathsf{Resp}(1^\lambda, x, w, (r, \mathsf{rnd}_P, \beta_1), \beta_2, \ldots, \beta_i)$:

1. Let $\hat{x} := (x, r)$ as in Chal and view w as a witness for $\hat{x} \in \widehat{\mathbf{L}}$.
2. Run the prover of $(\mathsf{rSWI.P}, \mathsf{rSWI.V})$ with prover input $(1^\lambda, \hat{x}, w)$, prover random tape rnd_P, and verifier messages $\beta_1, \ldots, \beta_i$. Let $\tilde{\alpha}_i$ be the resulting prover next message.
3. Output $\tilde{\alpha}_i$.

Figure 2: Our predictable argument $\mathsf{PA} = (\mathsf{Chal}, \mathsf{Resp})$.

Theorem 6 (restatement of Theorem 2)**.** *Assume the existence of one-way functions. Then, if there exists a resettable statistical witness-indistinguishable argument for all languages in* NP*, there also exists a witness encryption scheme for all languages in* NP.

As stated in Section 3.5, witness encryption schemes and predictable arguments are equivalent. Thus, to prove Theorem 6, it suffices to prove the following.

Theorem 7. *Assume the existence of one-way functions. Then, if there exists a resettable statistical witness-indistinguishable argument for all languages in* NP*, there also exists a predictable argument for all languages in* NP.

5.1 Proof of Theorem 7

Proof. Fix any NP language $\mathbf{L}$, and assume the existence of resettable statistical witness-indistinguishable arguments for all languages in NP. Our goal is to obtain a predictable argument for $\mathbf{L}$. Let PRG be any pseudorandom generator (whose existence is implied by the existence of one-way functions). For concreteness, we assume that PRG expands a λ-bit seed to a 2λ-bit pseudorandom string.

To obtain a predictable argument for $\mathbf{L}$, we use a resettable statistical witness-indistinguishable argument for a related NP language $\widehat{\mathbf{L}}$. Let $\mathbf{L}_{\mathsf{PRG}}$ be the NP language such that $\mathbf{L}_{\mathsf{PRG}} := \{r \mid \exists s \in \{0,1\}^\lambda \text{ s.t. } r = \mathsf{PRG}(s)\}$. Then, the NP language $\widehat{\mathbf{L}}$ is defined as follows.

$$\widehat{\mathbf{L}} = \{(x, r) \mid x \in \mathbf{L} \vee r \in \mathbf{L}_{\mathsf{PRG}}\}\ .$$

Let $\mathsf{rSWI} = (\mathsf{rSWI.P}, \mathsf{rSWI.V})$ be a resettable statistical witness-indistinguishable argument for $\widehat{\mathbf{L}}$. Let ρ be the round complexity of rSWI. Without loss of generality, we assume that the verifier sends the first message in rSWI.

Our predictable argument $\mathsf{PA} = (\mathsf{Chal}, \mathsf{Resp})$ for $\mathbf{L}$ is given in Figure 2. In what follows, we prove its completeness and soundness.

First, we prove the completeness of PA. From the construction of PA, it suffices to prove the following: When $(x, r) \in \widehat{\mathbf{L}}$ is chosen as in Chal, the resettable statistical witness-indistinguishable argument $(\mathsf{rSWI.P}, \mathsf{rSWI.V})$ produces the same transcript when it is used with the same randomness and two different witnesses (one is a witness w for $x \in \mathbf{L}$ and the other is a witness s for $r \in \mathbf{L}_{\mathsf{PRG}}$). Motivated by this observation, we consider the following lemma.

Lemma 1. *Let $\boldsymbol{L}$ and $\boldsymbol{L}_{\text{aux}}$ be* NP *languages. For each $\lambda \in \mathbb{N}$, let $D_{\text{aux},\lambda}$ be a distribution over $\boldsymbol{L}_{\text{aux}} \cap \{0,1\}^{\mathsf{poly}(\lambda)}$ that has negligible collision probability, i.e., $\Pr[x_0 = x_1 \mid x_0, x_1 \leftarrow D_{\text{aux},\lambda}] = \mathsf{negl}(\lambda)$. Let (P,V) be any resettable statistical witness-indistinguishable argument for $\widehat{\boldsymbol{L}} := \{(x,x') \mid x \in \boldsymbol{L} \vee x' \in \boldsymbol{L}_{\text{aux}}\}$. Then, there exists a negligible function* negl *such that for every $\lambda \in \mathbb{N}$, $x \in \boldsymbol{L}$, and $w \in \boldsymbol{R}_{\boldsymbol{L}}(x)$, it holds*

$$\Pr\left[\tau_0 \neq \tau_1 \;\middle|\; \begin{array}{l} x' \leftarrow D_{\text{aux},\lambda};\ w' \leftarrow \boldsymbol{R}_{\boldsymbol{L}_{\text{aux}}}(x');\ \hat{x} := (x,x') \\ \textit{Sample a random tape } \mathsf{rnd}_P \textit{ for } P \\ \textit{Sample a random tape } \mathsf{rnd}_V \textit{ for } V \\ \tau_0 := \mathsf{trans}\left[P(1^\lambda, \hat{x}, w; \mathsf{rnd}_P) \leftrightarrow V(1^\lambda, \hat{x}; \mathsf{rnd}_V)\right] \\ \tau_1 := \mathsf{trans}\left[P(1^\lambda, \hat{x}, w'; \mathsf{rnd}_P) \leftrightarrow V(1^\lambda, \hat{x}; \mathsf{rnd}_V)\right] \end{array}\right] \leq \mathsf{negl}(\lambda).$$

Given Lemma 1, we can prove the completeness of PA by observing that the sampling of $r \in \mathbf{L}_{\mathsf{PRG}} \cap \{0,1\}^{2\lambda}$ in PA has negligible collision probability because of the pseudorandomness of PRG. The proof of Lemma 1 is given in Section 5.2.

Next, we prove the soundness of PA. Let us denote the verifier of PA by PA.V (which interacts with the prover by using Chal as described in Definition 8). Assume for contradiction that there exists a PPT cheating prover P^* and polynomials poly, poly$'$ such that for infinitely many $\lambda \in \mathbb{N}$, there exist $x \in \{0,1\}^{\mathsf{poly}(\lambda)} \setminus \mathbf{L}$ and $z \in \{0,1\}^*$ such that

$$\Pr\left[\mathsf{output}_V\left[P^*(1^\lambda, x, z) \leftrightarrow V(1^\lambda, x)\right] = 1\right] \geq \frac{1}{\mathsf{poly}'(\lambda)}. \tag{9}$$

Fix any such λ, x, and z. We derive a contradiction by using a hybrid argument to break the soundness of (rSWI.P, rSWI.V) in the last hybrid experiment. Concretely, we consider the following hybrid experiments.

- Hybrid H_0 is the real experiment, where PA.V interacts with P^* by using Chal. Concretely, the interaction proceeds as follows.

  1. PA.V samples $(\boldsymbol{\beta}, \boldsymbol{\alpha}) \leftarrow \mathsf{Chal}(1^\lambda, x)$, where $\boldsymbol{\beta} := ((r, \mathsf{rnd}_P, \beta_1), \beta_2, \ldots, \beta_\rho)$ and $\boldsymbol{\alpha} := (\alpha_1, \ldots, \alpha_\rho)$.
  2. PA.V sends $(r, \mathsf{rnd}_P, \beta_1)$ to P^* and receives a reply $\tilde{\alpha}_1$ from P^*.
  3. For all $i \in \{2, \ldots, \rho\}$ in increasing sequence, PA.V sends β_i to P^* and receives a reply $\tilde{\alpha}_i$ from P^*.
  4. PA.V outputs 1 if and only if $\tilde{\alpha}_i = \alpha_i$ for all $i \in [\rho]$.

  From (9), PA.V outputs 1 with probability at least $1/\mathsf{poly}'(\lambda)$.

- Hybrid H_1 differs from H_0 in that in Step 4, PA.V outputs 0 if $(\beta_1, \tilde{\alpha}_1, \ldots, \beta_\rho, \tilde{\alpha}_\rho)$ is not an accepting transcript of (rSWI.P, rSWI.V).

  In this hybrid, PA.V outputs 1 with probability at least $1/\mathsf{poly}'(\lambda) - \mathsf{negl}(\lambda)$ because of the completeness of (rSWI.P, rSWI.V). Indeed, when $\tilde{\alpha}_i = \alpha_i$ for all $i \in [\rho]$, the transcript $(\beta_1, \tilde{\alpha}_1, \ldots, \beta_\rho, \tilde{\alpha}_\rho)$ is accepting with overwhelming probability since it is equal to the honest transcript $(\beta_1, \alpha_1, \ldots, \beta_\rho, \alpha_\rho)$ that Chal generated using a valid witness for $\hat{x} \in \widehat{\mathbf{L}}$.

- Hybrid H_2 differs from H_1 in that in Steps 2, 3, and 4, each verifier message β_i is replaced with the message $\tilde{\beta}_i$ that is computed by running the verifier of (rSWI.P, rSWI.V) with verifier input $\hat{x} = (x, r)$, verifier random tape rnd_V, and prover messages $(\tilde{\alpha}_1, \ldots, \tilde{\alpha}_{i-1})$, where rnd_V is the verifier random tape that was used in Chal to obtain $(\beta_1, \alpha_1, \ldots, \beta_\rho, \alpha_\rho)$.

  In this hybrid, PA.V still outputs 1 with probability at least $1/\mathsf{poly}'(\lambda) - \mathsf{negl}(\lambda)$ since each $\tilde{\beta}_i$ is equal to β_i when $\tilde{\alpha}_j = \alpha_j$ for every $j \in \{1, \ldots, i-1\}$.

- Hybrid H_3 differs from H_2 in that in Step 4, PA.V no longer checks $\tilde{\alpha}_i \stackrel{?}{=} \alpha_i$ for any $i \in [\rho]$ and outputs 1 if and only if $(\tilde{\beta}_1, \tilde{\alpha}_1, \ldots, \tilde{\beta}_\rho, \tilde{\alpha}_\rho)$ is an accepting transcript of (rSWI.P, rSWI.V).

  In this hybrid, PA.V still outputs 1 with probability at least $1/\mathsf{poly}'(\lambda) - \mathsf{negl}(\lambda)$ since the success probability of P^* only increases in this hybrid.

- Hybrid H_4 differs from H_3 in that in Step 1, (i) Chal is no longer executed, (ii) rnd_P and rnd_V are sampled uniformly as in Chal, and (iii) r is sampled uniformly from $\{0,1\}^{2\lambda}$. (That is, $\boldsymbol{\beta}$ and $\boldsymbol{\alpha}$ are no longer generated, and r is truly random rather than pseudorandom.)

  In this hybrid, $\mathsf{PA.V}$ outputs 1 with probability at least $1/\mathsf{poly}'(\lambda) - \mathsf{negl}'(\lambda)$ for a negligible function negl' because of the pseudorandomness of PRG.

We derive a contradiction by using H_4 to construct a successful cheating prover P^*_{rSWI} against the soundness of $(\mathsf{rSWI.P}, \mathsf{rSWI.V})$. In H_4, we have $(x,r) \notin \widehat{\mathbf{L}}$ with overwhelming probability since $r \in \{0,1\}^{2\lambda}$ is in the image of PRG with probability at most $1/2^{\lambda}$. Thus, from an average argument, we can fix r in H_4 in such a way that (i) $(x,r) \notin \widehat{\mathbf{L}}$ and (ii) with this fixed value, $\mathsf{PA.V}$ outputs 1 with non-negligible probability. For any such r, our cheating prover P^*_{rSWI} interacts with an honest verifier of $(\mathsf{rSWI.P}, \mathsf{rSWI.V})$ with statement (x,r) as follows.

> P^*_{rSWI} internally invokes P^* and executes H_4 while forwarding the messages from the external verifier to P^* as $\tilde{\beta}_1, \dots, \tilde{\beta}_\rho$ and forwarding the messages $\tilde{\alpha}_1, \dots, \tilde{\alpha}_\rho$ from P^* to the external verifier.

From the construction, P^*_{rSWI} perfectly emulates H_4 for the internal P^*. Thus, P^*_{rSWI} makes the external verifier output 1 with non-negligible probability. Since $\hat{x} = (x,r) \notin \widehat{\mathbf{L}}$, we have derived a contradiction. □

5.2 Proof of Lemma 1

Proof. Assume for contradiction that there exists a polynomial p such that for infinitely many $\lambda \in \mathbb{N}$, there exist $x_{\mathbf{L}} \in \mathbf{L}$ and $w_{\mathbf{L}} \in \mathbf{R}_{\mathbf{L}}(x_{\mathbf{L}})$ such that

$$\Pr\left[\tau_0 \neq \tau_1 \;\middle|\; \begin{array}{l} x' \leftarrow D_{\mathrm{aux},\lambda};\; w' \leftarrow \mathbf{R}_{\mathbf{L}_{\mathrm{aux}}}(x');\; \hat{x} := (x_{\mathbf{L}}, x') \\ \text{Sample a random tape } \mathsf{rnd}_P \text{ for } P \\ \text{Sample a random tape } \mathsf{rnd}_V \text{ for } V \\ \tau_0 := \mathsf{trans}\left[P(1^\lambda, \hat{x}, w_{\mathbf{L}}; \mathsf{rnd}_P) \leftrightarrow V(1^\lambda, \hat{x}; \mathsf{rnd}_V)\right] \\ \tau_1 := \mathsf{trans}\left[P(1^\lambda, \hat{x}, w'; \mathsf{rnd}_P) \leftrightarrow V(1^\lambda, \hat{x}; \mathsf{rnd}_V)\right] \end{array}\right] \geq \frac{1}{p(\lambda)}. \tag{10}$$

Fix any such $\lambda, x_{\mathbf{L}}, w_{\mathbf{L}}$. Let ℓ_P be the length of rnd_P and ℓ_V be the length of rnd_V. Let $t := 4p(\lambda) \cdot (\ell_P + \lambda)$. For simplicity, we use the following notation below: $\tau(\hat{x}, w, \mathsf{rnd}_P, \mathsf{rnd}_V)$ denotes the transcript of (P, V) generated with statement $\hat{x}$, witness w, prover random tape rnd_P, and verifier random tape rnd_V. That is,

$$\tau(\hat{x}, w, \mathsf{rnd}_P, \mathsf{rnd}_V) := \mathsf{trans}\left[P(1^\lambda, \hat{x}, w; \mathsf{rnd}_P) \leftrightarrow V(1^\lambda, \hat{x}; \mathsf{rnd}_V)\right].$$

Given this notation, we can write (10) as follows.

$$\Pr_{x', w', \mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau(\hat{x}, w_{\mathbf{L}}, \mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau(\hat{x}, w', \mathsf{rnd}_P, \mathsf{rnd}_V)\right] \geq \frac{1}{p(\lambda)}, \tag{11}$$

where the probability is taken over $x' \leftarrow D_{\mathrm{aux},\lambda}$, $w' \leftarrow \mathbf{R}_{\mathbf{L}_{\mathrm{aux}}}(x')$, $\hat{x} := (x_{\mathbf{L}}, x')$, $\mathsf{rnd}_P \leftarrow \{0,1\}^{\ell_P}$, and $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$.

We derive a contradiction by breaking the resettable statistical witness indistinguishability of (P, V). In particular, we give a t-tuple of instances $(\hat{x}_1, \dots, \hat{x}_t)$ and two t-tuples of witnesses $(\hat{w}_1^{(0)}, \dots, \hat{w}_t^{(0)})$, $(\hat{w}_1^{(1)}, \dots, \hat{w}_t^{(1)})$ such that (i) each $\hat{w}_i^{(0)}$ and $\hat{w}_i^{(1)}$ are valid witnesses for $\hat{x}_i \in \widehat{\mathbf{L}}$ and (ii) we can easily design a cheating verifier and a distinguisher that break the resettable statistical witness indistinguishability of (P, V) w.r.t. these tuples.

5.2.1 Instances $(\hat{x}_1, \dots, \hat{x}_t)$.

We define $(\hat{x}_1, \dots, \hat{x}_t)$ so that the following holds: $\hat{x}_1, \dots, \hat{x}_t$ are t distinct instances of $\widehat{\mathbf{L}} \cap \{0,1\}^{\mathsf{poly}(\lambda)}$ such that for each $\hat{x}_i$, the probability in the left-hand side of (11) is at least $1/2p(\lambda)$ under the condition that $\hat{x} = \hat{x}_i$. Below, we observe that there indeed exist such t instances. In particular, we obtain such t instances with non-zero probability by sampling sufficiently many instances of $\mathbf{L}_{\mathrm{aux}}$ from $D_{\mathrm{aux},\lambda}$. For each x' in the support of $D_{\mathrm{aux},\lambda}$, let $\delta(x')$ denote the probability in the left-hand side of (11) with $\hat{x}$ being fixed to $(x_{\mathbf{L}}, x')$. That is,

$$\delta(x') := \Pr_{w', \mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau((x_{\mathbf{L}}, x'), w_{\mathbf{L}}, \mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau((x_{\mathbf{L}}, x'), w', \mathsf{rnd}_P, \mathsf{rnd}_V)\right].$$

From (11) and an average argument, we have

$$\Pr_{x' \leftarrow D_{\mathrm{aux},\lambda}}\left[\delta(x') \geq \frac{1}{2p(\lambda)}\right] \geq \frac{1}{2p(\lambda)}. \tag{12}$$

Let N be the random variable representing the number of samples that we need to obtain from $D_{\mathrm{aux},\lambda}$ to obtain an instance x' such that $\delta(x') \geq 1/2p(\lambda)$. Then, from (12), the expected number of samples that we need to obtain from $D_{\mathrm{aux},\lambda}$ to obtain t such instances is at most $t \cdot \mathbb{E}[N] \leq t \cdot 2p(\lambda)$. Therefore, from Markov's inequality, when we sample $2t \cdot 2p(\lambda)$ instances of $\mathbf{L}_{\mathsf{aux}}$ by $x'_i \leftarrow D_{\mathrm{aux},\lambda}$ for each $i \in [2t \cdot 2p(\lambda)]$, we have

$$\Pr_{\forall i \in [2t \cdot 2p(\lambda)] : x'_i \leftarrow D_{\mathrm{aux},\lambda}}\left[\left|\left\{x'_i \text{ s.t. } \delta(x'_i) \geq \frac{1}{2p(\lambda)}\right\}\right| \geq t\right] \geq \frac{1}{2}. \tag{13}$$

Recalling that $D_{\mathrm{aux},\lambda}$ is assumed to satisfy $\Pr[x'_0 = x'_1 \mid x'_0, x'_1 \leftarrow D_{\mathrm{aux},\lambda}] = \mathsf{negl}(\lambda)$, we obtain the following from (13).

$$\Pr_{\forall i \in [2t \cdot 2p(\lambda)] : x'_i \leftarrow D_{\mathrm{aux},\lambda}}\left[\left|\left\{x'_i \text{ s.t. } \delta(x'_i) \geq \frac{1}{2p(\lambda)}\right\}\right| \geq t \wedge x'_i \neq x'_j \text{ for } \forall i \neq j\right]$$
$$\geq \frac{1}{2} - (2t \cdot 2p(\lambda))^2 \cdot \mathsf{negl}(\lambda) > 0.$$

That is, with non-zero probability, we obtain distinct t instances $x'_{i_1}, \ldots x'_{i_t} \in \mathbf{L}_{\mathrm{aux}} \cap \{0,1\}^{\mathsf{poly}(\lambda)}$ such that $\delta(x'_{i_j}) \geq 1/2p(\lambda)$ for every $j \in [t]$. By fixing any such $x'_{i_1}, \ldots x'_{i_t}$ and defining the t-tuple $(\hat{x}_1, \ldots, \hat{x}_t)$ by $\hat{x}_j := (x_{\mathbf{L}}, x'_{i_j})$ for every $j \in [t]$, we can guarantee that the t-tuple $(\hat{x}_1, \ldots, \hat{x}_t)$ satisfies the desired requirement.

5.2.2 Witnesses $(\hat{w}_1^{(0)}, \ldots, \hat{w}_t^{(0)})$.

Each $\hat{w}_i^{(0)}$ is an arbitrary witness for $\hat{x}_i \in \widehat{\mathbf{L}}$.

5.2.3 Witnesses $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$.

We define $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$ so that the following holds w.r.t. the above-defined $(\hat{x}_1, \ldots, \hat{x}_t)$ and $(\hat{w}_1^{(0)}, \ldots, \hat{w}_t^{(0)})$.

Requirement 1. *When sampling* $\mathsf{rnd}_P^{(1)} \leftarrow \{0,1\}^{\ell_P}$ *and* $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$*, we have*

$$\Pr_{\mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V}\left[\begin{array}{l} \nexists \mathsf{rnd}_P^{(0)} \in \{0,1\}^{\ell_P} \textit{ s.t. } \forall i \in [t]: \\ \tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau(\hat{x}_i, \hat{w}_i^{(1)}, \mathsf{rnd}_P^{(1)}, \mathsf{rnd}_V) \end{array}\right] \geq \frac{1 - 2^{-\lambda}}{4p(\lambda)}.$$

That is, the requirement is that when we generate t transcripts of (P, V) by using witnesses $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$ and (common) uniform randomness, with non-negligible probability they cannot be "explained" as being generated with witnesses $(\hat{w}_1^{(0)}, \ldots, \hat{w}_t^{(0)})$. Later, we use this requirement to design a distinguisher that breaks the resettable statistical witness indistinguishability of (P, V). (Essentially, the distinguisher checks whether the given t transcripts can be explained as being generated with $(\hat{w}_1^{(0)}, \ldots, \hat{w}_t^{(0)})$.) In what follows, we observe that there indeed exist t witnesses satisfying this requirement.

Before defining $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$, we make a preliminary observation. Roughly speaking, we observe that there are t witnesses $w'_1, \ldots, w'_t$ for $x'_1, \ldots, x'_t \in \mathbf{L}_{\mathsf{aux}}$ such that when sampling uniform random tapes rnd_P and rnd_V, the transcript generated with $w_{\mathbf{L}}$ and that generated with w'_i disagree at sufficiently many i's. Concretely, we observe the following. Recall that the t-tuple of instances $(\hat{x}_1, \ldots, \hat{x}_t)$ are defined so that for each $\hat{x}_i = (x_{\mathbf{L}}, x'_i)$, we have the following when sampling $w' \leftarrow \mathbf{R}_{\mathbf{L}_{\mathrm{aux}}}(x')$, $\mathsf{rnd}_P \leftarrow \{0,1\}^{\ell_P}$, and $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$.

$$\Pr_{w', \mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau(\hat{x}_i, w_{\mathbf{L}}, \mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau(\hat{x}_i, w', \mathsf{rnd}_P, \mathsf{rnd}_V)\right] \geq \frac{1}{2p(\lambda)}.$$

From an average argument, for each $\hat{x}_i = (x_{\mathbf{L}}, x'_i)$, there exists $w'_i \in \mathbf{R}_{\mathbf{L}_{\mathrm{aux}}}(x'_i)$ for which the above holds, i.e.,

$$\Pr_{\mathsf{rnd}_P, \mathsf{rnd}_V}\left[\tau(\hat{x}_i, w_{\mathbf{L}}, \mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau(\hat{x}_i, w'_i, \mathsf{rnd}_P, \mathsf{rnd}_V)\right] \geq \frac{1}{2p(\lambda)}. \tag{14}$$

For each $i \in [t]$, fix any w'_i satisfying (14). For each $i \in [t]$, $\mathsf{rnd}_P \in \{0,1\}^{\ell_P}$, and $\mathsf{rnd}_V \in \{0,1\}^{\ell_V}$, let $\tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)$ and $\tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)$ be the transcripts defined as follows.

$$\tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V) := \tau(\hat{x}_i, w_{\mathbf{L}}, \mathsf{rnd}_P, \mathsf{rnd}_V).$$
$$\tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V) := \tau(\hat{x}_i, w'_i, \mathsf{rnd}_P, \mathsf{rnd}_V).$$

Then, from (14) and the linearity of expectation, we have the following when sampling $\mathsf{rnd}_P \leftarrow \{0,1\}^{\ell_P}$ and $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$.

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\mathsf{rnd}_P,\mathsf{rnd}_V}\left[|\{i \in [t] \text{ s.t. } \tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\}|\right] \\
&= \sum_{i\in[t]} \Pr_{\mathsf{rnd}_P,\mathsf{rnd}_V}\left[\tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\right] \\
&\geq \frac{t}{2p(\lambda)}.
\end{aligned} \tag{15}$$

Thus, as stated at the beginning of this paragraph, a random transcript generated with $w_{\mathbf{L}}$ and that generated with w'_i disagree at many i's.

We proceed to define $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$. From (15) and an average argument, with probability at least $1/4p(\lambda)$ over the choice of $\mathsf{rnd}_P \leftarrow \{0,1\}^{\ell_P}$ and $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$, we have

$$|\{i \in [t] \text{ s.t. } \tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\}| \geq \frac{t}{4p(\lambda)} = \ell_P + \lambda. \tag{16}$$

Note that for any $\mathsf{rnd}_P \in \{0,1\}^{\ell_P}$ and $\mathsf{rnd}_V \in \{0,1\}^{\ell_V}$ satisfying (16), we have

$$\begin{aligned}
&\Pr_{\forall i\in[t]:b_i\leftarrow\{0,1\}}\left[\begin{array}{l}\exists \mathsf{rnd}_P^{(0)} \in \{0,1\}^{\ell_P} \text{ s.t. } \forall i \in [t]: \\ \tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_{b_i,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\end{array}\right] \\
&\leq \sum_{\mathsf{rnd}_P^{(0)}\in\{0,1\}^{\ell_P}} \Pr_{\forall i\in[t]:b_i\leftarrow\{0,1\}}\left[\begin{array}{l}\forall i \in [t]: \\ \tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_{b_i,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\end{array}\right] \\
&\leq 2^{\ell_P} \cdot \frac{1}{2^{\ell_P+\lambda}} = \frac{1}{2^\lambda}.
\end{aligned} \tag{17}$$

(The first inequality follows from the union bound. The second inequality follows from (16) since for any i such that $\tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V) \neq \tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)$, at least one of these two transcripts disagrees with $\tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V)$.) Recalling that we have (16) with probability at least $1/4p(\lambda)$ over the choice of rnd_P and rnd_V, we obtain the following from (17): When sampling $\mathsf{rnd}_P \leftarrow \{0,1\}^{\ell_P}$, $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$, and $b_i \leftarrow \{0,1\}$ for every $i \in [t]$, we have

$$\Pr_{\mathsf{rnd}_P,\mathsf{rnd}_V,b_1,\ldots,b_t}\left[\begin{array}{l}\nexists \mathsf{rnd}_P^{(0)} \in \{0,1\}^{\ell_P} \text{ s.t. } \forall i \in [t]: \\ \tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) \neq \tau_{b_i,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\end{array}\right] \geq \frac{1-2^{-\lambda}}{4p(\lambda)}. \tag{18}$$

From an average argument, we can fix $b_1, \ldots, b_t \in \{0,1\}$ in (18) so that when sampling $\mathsf{rnd}_P \leftarrow \{0,1\}^{\ell_P}$ and $\mathsf{rnd}_V \leftarrow \{0,1\}^{\ell_V}$, we have

$$\Pr_{\mathsf{rnd}_P,\mathsf{rnd}_V}\left[\begin{array}{l}\nexists \mathsf{rnd}_P^{(0)} \in \{0,1\}^{\ell_P} \text{ s.t. } \forall i \in [t]: \\ \tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) \neq \tau_{b_i,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)\end{array}\right] \geq \frac{1-2^{-\lambda}}{4p(\lambda)}. \tag{19}$$

Let us define $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$ by $\hat{w}_i^{(1)} := w_{\mathbf{L}}$ when $b_i = 0$ and $\hat{w}_i^{(1)} := w'_i$ when $b_i = 1$ for each $i \in [t]$. Then, recalling the definitions of $\tau_{0,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)$ and $\tau_{1,i}(\mathsf{rnd}_P, \mathsf{rnd}_V)$, we can see from (19) that the t-tuple $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$ satisfies Requirement 1 as desired.

5.2.4 Deriving a contradiction.

We are ready to derive a contradiction. Consider the following verifier and distinguisher against the resettable statistical witness indistinguishability of (P, V).

Verifier V^*. Recall that V^* interacts with deterministic prover strategies $P^{(1)}, \ldots P^{(t_\lambda)}$ that are defined by $P^{(i)}(\alpha) = P(1^\lambda, \hat{x}_i, \hat{w}_i^{(b)}, \alpha; \mathsf{rnd}_P)$ for each $i \in [t]$. (The prover random tape rnd_P and the choice b are unknown to V^*.)

V^* takes $z = ((\hat{x}_1, \ldots, \hat{x}_t), (\hat{w}_1^{(0)}, \ldots, \hat{w}_t^{(0)}), (\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)}))$ as non-uniform input. V^* uniformly samples a random tape $\mathsf{rnd}_V \in \{0,1\}^{\ell_V}$ for the honest verifier strategy V and interacts with each $P^{(i)}$ by using V with statement $\hat{x}_i$ and random tape rnd_V. Let $\tau_1, \ldots, \tau_t$ be the t resulting transcripts. Then, V^* outputs $(\tau_1, \ldots, \tau_t, \mathsf{rnd}_V, z)$.

Distinguisher D. D takes as input the verifier output $(\tau_1, \ldots, \tau_t, \mathsf{rnd}_V, z)$. Then, D checks by brute force whether there exists $\mathsf{rnd}_P^{(0)} \in \{0,1\}^{\ell_P}$ such that $\tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_i$ for every $i \in [t]$. If such $\mathsf{rnd}_P^{(0)}$ exists, D outputs 0. Otherwise, D outputs 1.

Let us analyze the success probability of D. When the deterministic prover strategies $P^{(1)}, \ldots P^{(t_\lambda)}$ use $(\hat{w}_1^{(0)}, \ldots, \hat{w}_t^{(0)})$, the distinguisher D never output 1 since there always exists $\mathsf{rnd}_P^{(0)} \in \{0,1\}^{\ell_P}$ such that $\tau(\hat{x}_i, \hat{w}_i^{(0)}, \mathsf{rnd}_P^{(0)}, \mathsf{rnd}_V) = \tau_i$ for every $i \in [t]$. When the deterministic prover strategies $P^{(1)}, \ldots P^{(t_\lambda)}$ use $(\hat{w}_1^{(1)}, \ldots, \hat{w}_t^{(1)})$, the distinguisher D outputs 1 with non-negligible probability because of Requirement 1. Thus, V^* and D successfully break the resettable statistical witness indistinguishability of (P, V). Therefore, we have derived a contradiction. □

6 Impossibility of Resettable Statistically Hiding Commitment

This section shows the impossibility of resettable statistically hiding commitment schemes.

Theorem 8. *There does not exist any statistically hiding commitment scheme.*

Proof. Fix any computationally binding commitment scheme (C, R). We show that (C, R) cannot be resettable statistically hiding. Let $\ell_C(\lambda)$ denote the length of the committer random tape and $\ell_R(\lambda)$ denote the length of the receiver randomness. Consider the following cheating receiver and distinguisher against the resettable statistical hiding of (C, R).

Receiver R^*. Recall that R^* is allowed to interact with deterministic committer strategy C_b polynomially many times, where C_b is defined by $C_b(\alpha) = C(1^\lambda, b, \alpha; \mathsf{rnd}_C)$. (The private input b and the committer random tape rnd_C are unknown to R^*.) Let $t(\lambda) := \ell_C(\lambda) + \lambda$.

For each $i \in [t(\lambda)]$ in sequence, R^* samples a random tape rnd_i for the honest receiver strategy R and interacts with C_b by using R with random tape rnd_i. Let $\tau_1, \ldots, \tau_{t(\lambda)}$ be the $t(\lambda)$ resulting transcripts. Then, R^* outputs $(\tau_1, \ldots, \tau_{t(\lambda)}, \mathsf{rnd}_1, \ldots, \mathsf{rnd}_{t(\lambda)})$.

Distinguisher D. D takes the receiver output $(\tau_1, \ldots, \tau_{t(\lambda)}, \mathsf{rnd}_1, \ldots, \mathsf{rnd}_{t(\lambda)})$ as input. Then, D checks by brute force whether there exists $\mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C(\lambda)}$ such that for every $i \in [t(\lambda)]$, it holds $\mathsf{trans}[C(1^\lambda, 0; \mathsf{rnd}_C^{(0)}) \leftrightarrow R(1^\lambda; \mathsf{rnd}_i)] = \tau_i$. If such $\mathsf{rnd}_C^{(0)}$ exists, D outputs 0. Otherwise, D outputs 1.

Let us analyze R^* and D. Fix any $\lambda \in \mathbb{N}$. For any $b \in \{0,1\}$, $\mathsf{rnd}_C \in \{0,1\}^{\ell_C(\lambda)}$, and $\mathsf{rnd}_R \in \{0,1\}^{\ell_R(\lambda)}$, let $\tau_b(\mathsf{rnd}_C, \mathsf{rnd}_R)$ be defined by

$$\tau_b(\mathsf{rnd}_C, \mathsf{rnd}_R) := \mathsf{trans}[C(1^\lambda, b; \mathsf{rnd}_C) \leftrightarrow R(1^\lambda; \mathsf{rnd}_R)].$$

To show that R^* and D indeed break the resettable statistical hiding of (C, R), it suffices to show the following for every $\mathsf{rnd}_C^{(1)} \in \{0,1\}^{\ell_C(\lambda)}$.

$$\Pr_{\forall i \in [t(\lambda)]: \mathsf{rnd}_i \leftarrow \{0,1\}^{\ell_R(\lambda)}} \left[\begin{array}{l} \exists \mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C(\lambda)} \text{ s.t. } \forall i \in [t(\lambda)] : \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_i) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_i) \end{array} \right] \leq \mathsf{negl}(\lambda). \tag{20}$$

(Indeed, if we have (20), the distinguisher D outputs 0 with negligible probability when $b = 1$ while it outputs 0 with probability 1 when $b = 0$.) Thus, we focus on showing (20). For any $\mathsf{rnd}_C^{(1)} \in \{0,1\}^{\ell_C(\lambda)}$, we use the union bound to obtain

$$\begin{aligned}
&\Pr_{\forall i \in [t(\lambda)]:\mathsf{rnd}_i \leftarrow \{0,1\}^{\ell_R(\lambda)}}\left[\begin{array}{l}\exists \mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C(\lambda)} \text{ s.t. } \forall i \in [t(\lambda)]: \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_i) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_i)\end{array}\right] \\
&\leq \sum_{\mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C(\lambda)}} \Pr_{\forall i \in [t(\lambda)]:\mathsf{rnd}_i \leftarrow \{0,1\}^{\ell_R(\lambda)}}\left[\begin{array}{l}\forall i \in [t(\lambda)]: \\ \tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_i) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd}_i)\end{array}\right] \\
&\leq \sum_{\mathsf{rnd}_C^{(0)} \in \{0,1\}^{\ell_C(\lambda)}} \left(\Pr_{\mathsf{rnd} \leftarrow \{0,1\}^{\ell_R(\lambda)}}\left[\tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd})\right]\right)^{t(\lambda)}.
\end{aligned} \tag{21}$$

Note that from the computational binding of (C, R) (which we assume to hold against non-uniform adversaries), we have the following for every $\mathsf{rnd}_C^{(0)}, \mathsf{rnd}_C^{(1)} \in \{0,1\}^{\ell_C(\lambda)}$.

$$\Pr_{\mathsf{rnd} \leftarrow \{0,1\}^{\ell_R(\lambda)}}\left[\tau_0(\mathsf{rnd}_C^{(0)}, \mathsf{rnd}) = \tau_1(\mathsf{rnd}_C^{(1)}, \mathsf{rnd})\right] \leq \mathsf{negl}(\lambda) \leq \frac{1}{2}. \tag{22}$$

(Indeed, if the above does not hold, a non-uniform cheating committer can break the binding property of (C, R) by committing to 0 using $\mathsf{rnd}_C^{(0)}$ and opening it to 1 using $\mathsf{rnd}_C^{(1)}$.) By combining (21) and (22) while recalling $t(\lambda) = \ell_C(\lambda) + \lambda$, we obtain (20) as desired. □

References

[ACM+17] Per Austrin, Kai-Min Chung, Mahmoody Mohammad, Rafael Pass, and Karn Seth. On the impossibility of cryptography with tamperable randomness. *Algorithmica*, 79:1052–1101, December 2017.

[BC20] Nir Bitansky and Arka Rai Choudhuri. Characterizing deterministic-prover zero knowledge. In Rafael Pass and Krzysztof Pietrzak, editors, *TCC 2020, Part I*, volume 12550 of *LNCS*, pages 535–566. Springer, Heidelberg, November 2020.

[BGGL01] Boaz Barak, Oded Goldreich, Shafi Goldwasser, and Yehuda Lindell. Resettably-sound zero-knowledge and its applications. In *42nd FOCS*, pages 116–125. IEEE Computer Society Press, October 2001.

[BKS21] Nir Bitansky, Michael Kellner, and Omri Shmueli. Post-quantum resettably-sound zero knowledge. In Kobbi Nissim and Brent Waters, editors, *TCC 2021, Part I*, volume 13042 of *LNCS*, pages 62–89. Springer, Heidelberg, November 2021.

[BOV12] Joshua Baron, Rafail Ostrovsky, and Ivan Visconti. Nearly simultaneously resettable black-box zero knowledge. In Artur Czumaj, Kurt Mehlhorn, Andrew M. Pitts, and Roger Wattenhofer, editors, *ICALP 2012, Part I*, volume 7391 of *LNCS*, pages 88–99. Springer, Heidelberg, July 2012.

[BP15] Nir Bitansky and Omer Paneth. On non-black-box simulation and the impossibility of approximate obfuscation. *SIAM Journal on Computing*, 44(5):1325–1383, 2015.

[CGGM00] 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.

[COP+14] Kai-Min Chung, Rafail Ostrovsky, Rafael Pass, Muthuramakrishnan Venkitasubramaniam, and Ivan Visconti. 4-round resettably-sound zero knowledge. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 192–216. Springer, Heidelberg, February 2014.

[COPV13] Kai-Min Chung, Rafail Ostrovsky, Rafael Pass, and Ivan Visconti. Simultaneous resettability from one-way functions. In *54th FOCS*, pages 60–69. IEEE Computer Society Press, October 2013.

[COSV12] Chongwon Cho, Rafail Ostrovsky, Alessandra Scafuro, and Ivan Visconti. Simultaneously resettable arguments of knowledge. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 530–547. Springer, Heidelberg, March 2012.

[COV17] Wutichai Chongchitmate, Rafail Ostrovsky, and Ivan Visconti. Resettably-sound resettable zero knowledge in constant rounds. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part II*, volume 10678 of *LNCS*, pages 111–138. Springer, Heidelberg, November 2017.

[CPS16] Kai-Min Chung, Rafael Pass, and Karn Seth. Non-black-box simulation from one-way functions and applications to resettable security. *SIAM Journal on Computing*, 45(2):415–458, 2016.

[CPW20] Suvradip Chakraborty, Manoj Prabhakaran, and Daniel Wichs. Witness maps and applications. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020, Part I*, volume 12110 of *LNCS*, pages 220–246. Springer, Heidelberg, May 2020.

[DGS09] Yi Deng, Vipul Goyal, and Amit Sahai. Resolving the simultaneous resettability conjecture and a new non-black-box simulation strategy. In *50th FOCS*, pages 251–260. IEEE Computer Society Press, October 2009.

[DL07] Yi Deng and Dongdai Lin. Instance-dependent verifiable random functions and their application to simultaneous resettability. In Moni Naor, editor, *EUROCRYPT 2007*, volume 4515 of *LNCS*, pages 148–168. Springer, Heidelberg, May 2007.

[DL20] Hila Dahari and Yehuda Lindell. Deterministic-prover zero-knowledge proofs. Cryptology ePrint Archive, Report 2020/141, 2020. https://eprint.iacr.org/2020/141.

[DNS98] Cynthia Dwork, Moni Naor, and Amit Sahai. Concurrent zero-knowledge. In *30th ACM STOC*, pages 409–418. ACM Press, May 1998.

[DOPS04] Yevgeniy Dodis, Shien Jin Ong, Manoj Prabhakaran, and Amit Sahai. On the (im)possibility of cryptography with imperfect randomness. In *45th FOCS*, pages 196–205. IEEE Computer Society Press, October 2004.

[FNV17] Antonio Faonio, Jesper Buus Nielsen, and Daniele Venturi. Predictable arguments of knowledge. In Serge Fehr, editor, *PKC 2017, Part I*, volume 10174 of *LNCS*, pages 121–150. Springer, Heidelberg, March 2017.

[GGH+13] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th FOCS*, pages 40–49. IEEE Computer Society Press, October 2013.

[GGSW13] Sanjam Garg, Craig Gentry, Amit Sahai, and Brent Waters. Witness encryption and its applications. In Dan Boneh, Tim Roughgarden, and Joan Feigenbaum, editors, *45th ACM STOC*, pages 467–476. ACM Press, June 2013.

[GMOS07] Vipul Goyal, Ryan Moriarty, Rafail Ostrovsky, and Amit Sahai. Concurrent statistical zero-knowledge arguments for NP from one way functions. In Kaoru Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 444–459. Springer, Heidelberg, December 2007.

[GMW91] Oded Goldreich, Silvio Micali, and Avi Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(3):691–729, July 1991.

[GO94] Oded Goldreich and Yair Oren. Definitions and properties of zero-knowledge proof systems. *Journal of Cryptology*, 7(1):1–32, December 1994.

[Gol01] Oded Goldreich. *Foundations of Cryptography: Basic Tools*, volume 1. Cambridge University Press, Cambridge, UK, 2001.

[GOVW12] Sanjam Garg, Rafail Ostrovsky, Ivan Visconti, and Akshay Wadia. Resettable statistical zero knowledge. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 494–511. Springer, Heidelberg, March 2012.

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid A. Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28(4):1364–1396, 1999.

[HNO+09] Iftach Haitner, Minh-Huyen Nguyen, Shien Jin Ong, Omer Reingold, and Salil Vadhan. Statistically hiding commitments and statistical zero-knowledge arguments from any one-way function. *SIAM Journal on Computing*, 39(3):1153–1218, 2009.

[IOS97] Toshiya Itoh, Yuji Ohta, and Hiroki Shizuya. A language-dependent cryptographic primitive. *Journal of Cryptology*, 10(1):37–50, December 1997.

[JLS21] Aayush Jain, Huijia Lin, and Amit Sahai. Indistinguishability obfuscation from well-founded assumptions. In Samir Khuller and Virginia Vassilevska Williams, editors, *53rd ACM STOC*, pages 60–73. ACM Press, June 2021.

[JLS22] Aayush Jain, Huijia Lin, and Amit Sahai. Indistinguishability obfuscation from LPN over $\mathbb{F}_p$, DLIN, and PRGs in NC^0. In Orr Dunkelman and Stefan Dziembowski, editors, *EUROCRYPT 2022, Part I*, volume 13275 of *LNCS*, pages 670–699. Springer, Heidelberg, May / June 2022.

[Kiy20] Susumu Kiyoshima. Statistical concurrent non-malleable zero-knowledge from one-way functions. *Journal of Cryptology*, 33(3):1318–1361, July 2020.

[MR01] Silvio Micali and Leonid Reyzin. Soundness in the public-key model. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 542–565. Springer, Heidelberg, August 2001.

[MU10] Jörn Müller-Quade and Dominique Unruh. Long-term security and universal composability. *Journal of Cryptology*, 23(4):594–671, October 2010.

[Nao91] Moni Naor. Bit commitment using pseudorandomness. *Journal of Cryptology*, 4(2):151–158, January 1991.

[OOR+14] Claudio Orlandi, Rafail Ostrovsky, Vanishree Rao, Amit Sahai, and Ivan Visconti. Statistical concurrent non-malleable zero knowledge. In Yehuda Lindell, editor, *TCC 2014*, volume 8349 of *LNCS*, pages 167–191. Springer, Heidelberg, February 2014.

[OSV15] Rafail Ostrovsky, Alessandra Scafuro, and Muthuramakrishnan Venkitasubramaniam. Resettably sound zero-knowledge arguments from OWFs - the (semi) black-box way. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part I*, volume 9014 of *LNCS*, pages 345–374. Springer, Heidelberg, March 2015.

[PRS02] 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.

[PTW09] Rafael Pass, Wei-Lung Dustin Tseng, and Douglas Wikström. On the composition of public-coin zero-knowledge protocols. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 160–176. Springer, Heidelberg, August 2009.

[Tsa22] Rotem Tsabary. Candidate witness encryption from lattice techniques. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part I*, volume 13507 of *LNCS*, pages 535–559. Springer, Heidelberg, August 2022.

[VWW22] Vinod Vaikuntanathan, Hoeteck Wee, and Daniel Wichs. Witness encryption and null-IO from evasive LWE. In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part I*, volume 13791 of *LNCS*, pages 195–221. Springer, Heidelberg, December 2022.