Unprovable Security of 2-Message Zero Knowledge

Kai-Min Chung* Edward Lui Mohammad Mahmoody† Rafael Pass‡

June 16, 2021

Abstract

Goldreich and Oren (JoC'94) show that only languages in **BPP** have 2-message zero-knowledge arguments. In this note we consider weaker, *super-polynomial simulation* (SPS), notions of zero-knowledge. We present barriers to using black-box reductions for demonstrating soundness of 2-message protocols with efficient prover strategies satisfying SPS zero-knowledge. More precisely, if poly($T(n)$)-hard one-way functions exist for a super-polynomial $T(n)$, the following holds about 2-message efficient prover arguments over statements of length n.

- Black-box reductions cannot prove soundness of 2-message $T(n)$-simulatable arguments based on any polynomial-time intractability assumption, unless the assumption can be broken in polynomial time. This complements known 2-message quasi-polynomial-time simulatable arguments using a quasi-polynomial-time reduction (Pass'03), and 2-message exponential-time simulatable proofs using a polynomial-time reduction (Dwork-Naor'00, Pass'03).
- Back-box reductions cannot prove soundness of 2-message *strong* $T(n)$-simulatable arguments, even if the reduction and the challenger both can run in poly($T(n)$)-time, unless the assumption can be broken in poly($T(n)$) time. Strong $T(\cdot)$-simulatability means that the output of the simulator is indistinguishable also for poly($T(\cdot)$)-size circuits, with a negl($T(\cdot)$) indistinguishability gap. This complements known 3-message strong quasi-polynomial-time simulatable proofs (Blum'86, Canetti et al' 00), or 2-message quasi-polynomial-time simulatable arguments (Khurana-Sahai'17, Kalai-Khurana-Sahai'18) satisfying a relaxed notion of strong simulation where the distinguisher's size can be large, but the distinguishing gap is negligible in n.

Contents

*Academia Sinica, kmchung@iis.sinica.edu.tw
†University of Virginia, mohammad@virginia.edu
‡Cornell University and Cornell Tech, rafael@cs.cornell.edu

1 Introduction

The notion of *zero-knowledge*, and the *simulation-paradigm* used to define it, is of fundamental importance in modern cryptography—most definitions of security rely on it. In a zero-knowledge protocol, a prover P can convince a verifier V of the naturality of some mathematical statement $x \in L$, while revealing "zero (additional) knowledge" to V. This zero-knowledge property is formalized by requiring that for every potentially malicious efficient verifier V^*, there exists an efficient simulator S that, without talking to P, is able to "indistinguishably reconstruct" the view of V^* in a true interaction with P. Namely, the output of S cannot be distinguished (with more than negligible probability) from the true view of V^* by any efficient distinguisher D.

Assuming standard cryptographic hardness assumptions, 3-message zero-knowledge proofs with constant soundness [Blu86], 4-message zero-knowledge *arguments* (where the soundness is guaranteed to hold only against *efficient* provers) with negligible soundness [FS90], and 5-message zero-knowledge proofs with negligible soundness [GK96] are known for all languages in **NP**; additionally, given a witness to the prover these interactive proofs/arguments have efficient prover strategies. On the other hand, by the results of Goldreich and Oren [GO94], 2-message zero-knowledge arguments only exist for languages in **BPP**. In the rest of this note, we focus on interactive proofs/arguments with negligible soundness error and with efficient prover strategies once a witness is given.

Super-Polynomial-Simulation (SPS) Zero-Knowledge. The commonly used notion of zero-knowledge requires the simulator to run in polynomial time. However, the notion of *super-polynomial-simulation (SPS) zero-knowledge* [Pas03] allows the simulator to run in super-polynomial time[1]. More specifically, the notion of *SPS zero-knowledge* is defined similarly to zero-knowledge except that the simulator is allowed to run in super-polynomial time $T(\cdot)$; such protocols are referred to as $T(\cdot)$*-simulatable*. [Pas03] also defined the (stronger) notion of *strong* SPS zero-knowledge with the additional requirement that any $\mathrm{poly}(T(\cdot))$-time distinguisher cannot distinguish the simulated transcript from a true transcript with better than $\mathrm{negl}(T(\cdot))$ advantage; such protocols are referred to as *strong* $T(\cdot)$*-simulatable*.

It is known that under *sub-exponential* hardness assumptions 2-message quasi-polynomial-time (i.e., $T(n) = n^{\mathrm{poly}(\log n)}$) simulatable *arguments* for **NP** exist, but 2-message $T(\cdot)$-simulatable *proofs* only exist for languages in $\mathbf{BPTime}(\mathrm{poly}(T(\cdot)))$ [Pas03]. On the other hand, based on sub-exponential hardness assumptions quasi-polynomial-time simulatable 3-message proofs for **NP** exist as well [Blu86, CGGM00].

This leaves open the following questions regarding 2-message SPS zero-knowledge:

1. *Do 2-message SPS zero-knowledge arguments for* **NP** *exist based on standard polynomial-time hardness assumptions?*
2. *Do 2-message strong SPS zero-knowledge arguments for* **NP** *exist even under super-polynomial hardness assumptions?*

In this note, we present barriers to using black-box reductions for providing affirmative answers to the above two questions. In particular, we show the following:

Theorem 1.1 (Informally Stated)**.** *Assuming the existence of* $\mathrm{poly}(T(n))$*-hard one-way functions, the following holds about efficient-prover arguments over statements of length* n*.*

[1] This is while the distinguisher running time remains polynomial time and the distinguishing advantage needs to be $\mathrm{negl}(n)$ which is non-negligible over the security parameter n rather than $\mathrm{negl}(T(n))$.

1. *Polynomial-time black-box reductions cannot be used to prove soundness of 2-message $T(n)$-simulatable arguments based on any intractability assumption is modeled as a security game with a polynomial-time challenger, unless the security game can be broken in time* $\mathrm{poly}(n)$.

2. *Even* $\mathrm{poly}(T(n))$*-time black-box reductions cannot be used to prove soundness of 2-message* strong $T(n)$*-simulatable arguments based on some security game, unless the security game can be broken in time* $\mathrm{poly}(T(n))$. *This holds even if we allow the challenger of the intractability assumption to run in* $\mathrm{poly}(T(n))$ *time.*

Let us remark here that n is the security parameter on which the black-box reduction calls the attacker (i.e., the statement length). The security reduction from a security game using parameter n' could also be calling the attacker A on a *polynomially smaller* security parameter n: for instance, consider an interactive argument of a statement of length n that uses a one-way function on input length $n' = n^c$; we refer to c as the security parameter blow-up. Whenever there is such a security parameter blow-up $c > 1$, we have that the security game given security parameter n' can be broken in time $\mathrm{poly}(T'(n'))$ where $T'(n) = T(n'^{1/c})$. When there is such a security parameter blow-up, Part 2 shows that for every ϵ, and every reduction R, there exists some $\epsilon' = \epsilon/c$ such that strong 2^{n^ϵ}-simulatable arguments cannot be based based on $2^{n^{\epsilon'}}$ hardness. It, however, does not rule out the existence of some $\epsilon'' > \epsilon'$ such that soundness can be based on $2^{n^{\epsilon''}}$ hardness.

The first part of our theorem complements known 2-message quasi-polynomial-time simulatable arguments using a quasi-polynomial-time reduction [Pas03] and 2-message exponential-time simulatable proofs using a polynomial-time reduction [DN07, Pas03]. The second part of our theorem complements (in terms of the round-complexity) the 3-message strong quasi-polynomial-time simulatable proofs of [Blu86, CGGM00]. Furthermore the second part of the theorem above also complements the results shown in [KS17, KKS18][2] showing that a *relaxed* form of strong quasi-polynomial-time simulatable 2-message arguments can be obtained when relaxing the notion of strong simulation to only requiring the distinguishing probability to be $\mathrm{negl}(n)$, as opposed to $\mathrm{negl}(T(n))$ (even when considering $T(n)$-time distinguishers).

Applications to the Soundness of the Fiat-Shamir Heuristic. The concurrent works of [DSJKLA12, BDSG+13] demonstrating barriers to provable security of the Fiat-Shamir heuristic also obtained similar results to the *second* part of Theorem 1.1 using similar techniques, and applied it to conclude as corollary that the Fiat-Shamir heuristic cannot be based on provable assumptions. Our result is incomparable to [DSJKLA12]. Our impossibility is proved for "natural" settings where the security reduction starts from an adversary who breaks the security for a particularly security parameter (which is the case in all black-box security reductions that we are aware of) while [DSJKLA12] allows the reduction to call the adversary on multiple security parameters. On the other hand, we can handle security reductions that assume the adversary to be fixed and deterministic, and we can handle a certain nonuniform techniques as well. Thus, as corollary of our Theorem 1.1, we can obtain an incomparable impossibility result for a general instantiation of the Fiat-Shamir heuristic for interactive proofs (which also deals with nonuniform reductions and deterministic attackers, whereas the results of [DSJKLA12] only rule out uniform reductions that need to work for randomized attackers.)[3] To see this, as we mentioned above, [CGGM00] shows (assuming one-way permutations with sub-exponential hardness) the existence of a 3-message strong

[2]These results were proved subsequent to the original appearance of this paper.

[3]We emphasize that this corollary was added after becoming aware of the results in [DSJKLA12].

quasi-polynomial-time simulatable proof (with negligible soundness error); additionally, this protocol is public coin. Assuming the soundness of the Fiat-Shamir heuristic (when applied only to proof systems), this 3-message proof system can be collapsed to a 2-message strong quasi-polynomial-time simulatable argument system (the "collapsed" protocol is still strong quasi-polynomial-time simulatable since the hash-function used in the Fiat-Shamir heuristic can just be viewed as a particular malicious verifier). Our Theorem 1.1 shows that this 2-message argument can not be proven sound through a black-box reduction to any "standard" assumption, and thus rules out a black-box reduction for a general instantiation of the Fiat-Shamir heuristic for interactive proofs. In contrast to these results, we note that a very recent beautiful line of work shows how to instantiate the Fiat-Shamir heuristic for special cases of interactive proofs [CCH+19, CLW18, PS19].

1.1 Further Related Work on Separations

There is a large literature on separation results between cryptographic primitives/assumptions. We distinguish between two types of results:

Separations for Fully Black-Box Constructions. The seminal work of Impagliazzo and Rudich [IR88] provides a framework for proving black-box separations between cryptographic primitives. We highlight that this framework refutes the possibility of so-called "fully-black-box constructions" (see [RTV04] for a taxonomy of various black-box separations); that is, this framework considers both black-box *constructions* (i.e., the higher-level primitive only uses the underlying primitive as a black-box), and black-box *proofs of security* (i.e., the security reduction only uses the adversary against the constructed scheme as a black-box). Most black-box separation results fall into this framework (e.g., [Sim98, GKM+00, BMG07, HHRS07] to name a few). As it was shown by [RTV04], some of these separations extend to the setting where the security reduction is "semi" or even "weakly" black-box, but we emphasize that the construction is always black-box.

Separations for Black-Box Security Reductions. In recent years, new types of separations between cryptographic primitives/assumptions have emerged. These separations apply even to non-black-box constructions as long as the proof of security is black-box: Pass [Pas06] and Pass, Tseng and Venkitasubramaniam [PTV11] demonstrate that under certain (new) complexity theoretic assumptions, various cryptographic tasks cannot be based on *one-way functions* using a black-box security reduction, even if the protocol uses the one-way function in a non-black-box way. (These results follow techniques used by Brassard [Bra83] and Akavia et al [AGGM06] to demonstrate limitations of "NP-hard cryptography".)[4]

Recently, two independent works demonstrate similar types of separation results, but this time ruling out security reductions to a *general* set of intractability assumptions: Pass [Pas11] demonstrates impossibility of using black-box reductions to prove the security of several primitives (e.g., Schnorr's identification scheme, commitment schemes secure under weak notions of selective opening, Chaum blind signatures, etc.) based on any "bounded-round" intractability assumption (where the challenger uses an a-priori bounded number of messages, but is otherwise unbounded). Gentry and Wichs [GW11] (assuming the existence of strong pseudorandom generators) demonstrate impossibility of using black-box security reductions to prove soundness of "succinct non-interactive

[4]See also the results of Feigenbaum and Fortnow [FF93] and the result of Bogdanov and Trevisan [BT03] that demonstrate limitations of NP-hard cryptography for *restricted* types of reductions.

arguments" based on any falsifiable assumption (where the challenger is computationally bounded). An even more recent work by Pass [Pas12], developed in parallel with the current paper, rules out constructions of statistical NIZK with adaptive soundness and non-interactive non-malleable commitments, based on falsifiable assumptions.

Our results in this work fall into this second category of results and rule out black-box security reductions for proving the soundness of various forms of SPS zero-knowledge protocols even if the construction is arbitrarily non-black-box.

2 Definitions

By U_n we denote the uniform distribution over $\{0,1\}^n$. We say $p = \text{poly}(n)$ if $p \leq n^{O(1)}$. We say $\nu(\cdot)$ is a negligible function, if for every $p = \text{poly}(n)$, there is n_0 such that $\nu(n) < p(n)$ for all $n > n_0$. By $\text{negl}(\cdot)$ we denote a negligible function. By $x \leftarrow S$ we denote that x is sampled uniformly from the set S. A verifier V_L for language $L \in \mathbf{NP}$ runs in polynomial time over $|x|$, and $x \in L$ iff $\exists w, V_L(x,w) = 1$, in which case w is called a witness for x (with respect to the verifier V_L). For an **NP** language L, the witness relation R_L contains the set of all (x,w) where w is a witness of membership of x in L.

We call a primitive $T(n)$-hard if for any $\text{poly}(T(n))$-size adversary, there is a negligible function ν such that adversary's advantage in breaking the primitive is at most $\nu(T(n))$ where n is the security parameter. This notion can be applied to one-way functions and (length-doubling) PRGs.

2.1 Zero-Knowledge Arguments

We recall the definition of interactive proofs/arguments and SPS-ZK.

Notation. For a pair of probabilistic interactive algorithms (P,V), $\langle P(y), V\rangle(x)$ denotes the output of V at the end of the interaction on the common input x. In the following, the term *polynomial time* always means polynomial time in the length of the first input. In particular, for an interactive protocol's execution $\langle P(y), V^*(z)\rangle(x)$ on common input x, x is the "first input" given to both parties.

Definition 2.1 (Interactive Arguments [GMR89, BCC88])**.** A pair of probabilistic interactive algorithms (P,V) is said to be an *interactive proof system for an* **NP***-language L with witness relation R_L* if V is PPT and the following two conditions hold:

- Completeness: For every $x \in L$ and every $y \in R_L(x)$, it holds that

$$\Pr\left[\langle P(y), V\rangle(x) = 1\right] = 1.$$

- Soundness: We say that P^* *breaks soundness of* (P,V) with probability $\mu(n)$ if

$$\Pr\left[\ (x,z) \leftarrow P^*(1^n)\ :\ \langle P^*(1^n,x,z), V(x)\rangle = 1 \wedge x \notin L\ \right] \geq \mu(n).$$

  We call (P,V) *sound*, if for every polynomial-time interactive algorithm P^*, if P^* breaks soundness of (P,V) with probability $\mu(n)$, then $\mu(n)$ is negligible in n. We say that (P,V) has an *efficient prover* if P is PPT.

We now give the definition of $T(n)$-simulatability.

Definition 2.2 ($T(n)$-Simulatability [Pas03])**.** Let (P, V) be an interactive proof/argument system for an **NP**-language L with witness relation R_L. We say that (P, V) is $T(n)$-*simulatable* if for every PPT adversary V^*, there exists a $T(n)$-time simulator S such that for every poly(n)-sized distinguisher D, there exists a negligible function $\nu(n)$ such that for every $x \in L$, $y \in R_L(x)$, and $z \in \{0,1\}^*$, it holds that

$$|\Pr[D(x, \langle P(y), V^*(z)\rangle(x)) = 1] - \Pr[D(x, S(x, z)) = 1]| \leq \nu(|x|).$$

We now give the definition of strong $T(n)$-simulatability.

Definition 2.3 (Strong $T(n)$-Simulatability [Pas03])**.** Let (P, V) be an interactive proof/argument system for an **NP**-language L with witness relation R_L. We say that (P, V) is strong $T(n)$-*simulatable* if for every PPT adversary V^*, there exists a $T(\cdot)$-time simulator S such that for every probabilistic poly$(T(\cdot))$-size distinguisher D, there exists a negligible function $\nu(n)$ such that for every $x \in L$, $y \in R_L(x)$, and $z \in \{0,1\}^*$,

$$|\Pr[D(x, \langle P(y), V^*(z)\rangle(x)) = 1] - \Pr[D(x, S(x, z)) = 1]| \leq \nu(T(|x|)).$$

The notions of *SPS zero-knowledge* and *strong SPS zero-knowledge* correspond, respectively, to $T(n)$-simulatability and strong $T(n)$-simulatability for a super-polynomial function $T(n)$. It is shown in [Pas03] that both plain and strong poly$(T(n))$-simulatability is closed under sequential composition; we will rely on the proof of this result.

2.2 Intractability Assumptions and Black-Box Reductions

Following Naor [Nao03] (see also [DOP05, HH09, RV10, GW11, Pas11]), we model an intractability assumption based on an interactive game between a probabilistic machine C—called the challenger—and an attacker A. Both parties get as input 1^n where n is the security parameter. For any $t(n) \in [0,1]$ and any "adversary" A, if $\Pr[\langle A, C\rangle(1^n) = 1] \geq t(n) + p(n)$, then we say that A *breaks* C *with advantage* $p(n)$ over the "threshold" $t(n)$. When this happens, we might also say that A *breaks* $(C, t(\cdot))$ *with advantage* $p(n)$. For the simple case of polynomial-time adversaries, any pair $(C, t(\cdot))$ intuitively corresponds to the following assumption.

> **Assumption** $(C, t(\cdot))$**:** *For any polynomial-time adversary A, there exists a negligible function $\nu(\cdot)$ such A breaks C with advantage at most $\nu(n)$ over the threshold $t(n)$.*

More generally, one can use $(C, t(\cdot))$ to model an assumption about adversaries with more resources (e.g., quasi-polynomial-time adversaries). This aspect becomes important once we define *reductions* of security to hardness assumptions, where the running time of the reduction is limited by the corresponding complexity class of the adversary in the intractability assumption.

If the challenger C of the assumption $(C, t(\cdot))$ is polynomial-time in the security parameter n' and the total length of the messages it receives, then we say that the assumption is *efficient challenger*; such assumptions are referred to as *falsifiable* assumptions by Naor [Nao03] and Gentry and Wichs [GW11]. $(C, t(\cdot))$ is an efficient challenger assumption if and only if $(C, t(\cdot))$ has a polynomial-time (or size) challenger. More generally, we can allow the challenger in $(C, t(\cdot))$ to run in super-polynomial $T'(n')$ where n' is the security parameter of $(C, t(\cdot))$. Note that we can capture relying on super-polynomial hardness assumptions by allowing for reductions to an assumption that run in super-polynomial-time over the security parameter of the challenge $(C, t(\cdot))$.

Black-Box Reductions. We consider PPT Turing reductions—i.e., *black-box reductions*. A black-box reduction refers to a PPT oracle algorithm. Roughly speaking, a black-box reduction for basing the security of a P on the hardness of an assumption $(C, t(\cdot))$, is a PPT oracle machine R such that whenever the oracle A "breaks" P with respect to the security parameter n, then R^A "breaks" $(C, t(\cdot))$ with respect to a polynomially related parameter n' such that n' is at most polynomially bigger than n. As far as we know, *all* security reductions fall into this framework. (See the discussion after Definition 2.4.)

Definition 2.4 (Natural Reductions)**.** We say that R is a *natural* black-box reduction to breaking $\mathcal{C} = (C, t(\cdot))$, if R is an oracle machine such that the following holds:

- **Relation between Security Parameters.** There exists some constant c such that if $R(1^{n'})$ ever queries its oracle on security parameter n, then $n' \leq n^c$; we refer to c as the *security parameter blow-up* of the reduction.
- **Single Adversary Security Parameter.** For every security parameter n', there exists some fixed security parameter n such that $R(1^{n'})$ only invokes its oracle on security parameter n.

Note that the definition of natural reductions above applies even if R is a reduction to a *super-polynomial* $T'(\cdot)$-hardness assumption, and it might be that n is supper-polynomially larger than n' (but not the other way around). In particular, suppose we are using a natural reduction to a $\mathrm{poly}(T'(\cdot))$-hard assumption. Then, the reduction's running time should be bounded as $T(n) \leq \mathrm{poly}(T'(n'))$, because we would like the security reduction to run in time $\mathrm{poly}(T'(\cdot))$.

Discussion. Definition 2.4 places some restrictions on the reduction, but as far as we know (for all the reasons explained below) all black-box security reductions are natural. Below we explain some of the reasons that make security reductions in the literature natural.

Single Adversary Security Parameter. The reason to restrict R to only query its oracle on a single "security parameter" n (which is the case also in all known security reductions in the literature), is that standard cryptographic definitions require ruling out the existence of attackers that break some primitive even for *any* infinite sequence of input lengths; as these input lengths can be very sparse, a black-box reduction might only get to access the adversary over a single "good" input length. Therefore, it must successfully use the adversary even if it has access to an attacker that only succeeds on a single input length. We formalize this argument in the Appendix A; see Lemma A.5.

Relation between Security Parameters. The reason to assume $n' \leq \mathrm{poly}(n)$ in the security reduction is the following. Suppose we have constructed a primitive on security parameter n using computationally hard "puzzless" (e.g., one-way functions or public-key encryption) on security parameter n'. Due to the polynomial running time of the *scheme* itself, it would be the case that $n' \leq \mathrm{poly}(n)$ where n is the security parameter of the scheme. Furthermore, the security reduction in its most natural form (which includes the actual reductions that we are aware of), would relate the security of the scheme to the security of one of puzzle used in it. Then, in the security reduction it would hold that the security parameter n of the adversary breaking the scheme (which is the same security parameter as the scheme itself) satisfies $n' \leq \mathrm{poly}(n)$ where n' is the challenge the reduction tries to break. This holds even if one uses *sub-exponentially* hard variant of these puzzles, in which case we might have $n' \ll n$ (e.g., $n' = \mathrm{polylog}(n)$).

Let us turn to defining what it means for a reduction to be successful. For concreteness, we directly define it for the specific security game of that we focus on in this paper: soundness of a 2-message arguments.

Definition 2.5 (Natural Reductions for Soundness of 2-Message Arguments)**.** Let $(C, t(\cdot))$ be an assumption and (P, V) a 2-message argument. We call R a *natural reduction for basing soundness of* (P, V) *on* $T'(\cdot)$*-hardness of the challenge* $(C, t(\cdot))$ if R is a natural reduction and there exists some inverse polynomial function α such that the following holds. Consider some n, n' such that $R^A(1^{n'})$ invokes its attacker on security parameter n, and let A be an arbitrary deterministic attacker that breaks the soundness of (P, V) with probability at least $1/2$ on security parameter (i.e., statement length) n. Then $R^A(1^{n'})$ breaks $(C, t(\cdot))$ over security parameter n' with probability $t(n') + \alpha(T'(n'))$, while running in time at most $\text{poly}(T'(n'))$.

Note that the default definition of soundness requires the error to be negligible, however in the definition above we use $1/2$ as the threshold for success probability for the adversary, because our goal is to prove impossibility results, and this choice makes the result only stronger. Also note that we only consider *deterministic* attackers; this only makes our result stronger, as we automatically get a stronger separation for *weakly nonuniform* security reductions as well.

Weakly Nonuniform Reductions. Suppose R is a security reduction that uses a (potentially) randomized adversary A on security parameter n' and breaks a challenge $(C, t(\cdot))$ on security parameter n. A technique commonly used in the literature is to allow the reduction to fix the randomness of A to its "best" value (perhaps to fix some messages sent form A). Using this technique is justified as, having access to such A, leads a an efficient *circuit* that breaks $(C, t(\cdot))$. Hence, if one can start from a stronger *nonuniform* hardness assumption about $(C, t(\cdot))$, the conclusion is that no such adversary A exists. More formally, to capture this technique, we define *weakly nonuniform reductions* as follows (again focusing on the special case of soundness for 2 message arguments).

Definition 2.6 (Weakly Nonuniform Natural Reductions for Soundness of 2-Message Arguments)**.** Let $(C, t(\cdot))$ be an assumption and (P, V) a 2-message argument. We call R a *weakly nonuniform natural* reduction for basing soundness of (P, V) on $T'(\cdot)$-hardness of $(C, t(\cdot))$ if R satisfies Definition 2.5 with the exception that in the second bullet, we require the existence of some randomized (potentially unbounded) function F such that for any *randomized* attacker A that breaks the soundness of (P, V) with probability at least $1/2$ for security parameter n, $R^{A_{F(A)}}(1^{n'})$ breaks $(C, t(\cdot))$ with probability $t(n') + \alpha(T'(n'))$, where A_r denotes A with randomness fixed to r.

We now observe that Definition 2.6 implies Definition 2.5; this result crucially relies on the fact that in Definition 2.5 we work with *deterministic* adversaries. We state the lemma for the specific case of reducing the soundness of 2-message arguments to black-box assumptions, but a similar statement holds in general as well.

Lemma 2.7. *Let R be a weakly nonuniform natural reduction for basing soundness of (P, V) on $T'(\cdot)$-hardness of $(C, t(\cdot))$. Then R is also a natural reduction for basing soundness of (P, V) on $T'(\cdot)$-hardness of $(C, t(\cdot))$.*

Proof. Let R be a weakly nonuniform natural reduction for basing soundness of (P, V) on $T'(\cdot)$-hardness of $(C, t(\cdot))$, F be the associated randomness fixing function, and $\alpha(\cdot)$ be the "advantage". We will now show that R is also a successful *natural* reduction. Consider some n, n' such that

$R(1^{n'})$ calls its attacker on security parameter n. Recall that a natural reduction only needs to be successful for *deterministic* attackers; thus, consider some deterministic attacker A such that A breaks soundness of (P, V) with probability $1/2$ given security parameter n. Since R is a successful weakly nonuniform reduction, we have that $R^{A_{F(A)}}(1^{n'})$ breaks $(C, t(\cdot))$ with probability $t(n') + \alpha(T'(n'))$. But since A is deterministic, $A = A_{F(A)}$, and thus we have that $R^{A}(1^{n'})$ breaks $(C, t(\cdot))$ with probability $t(n') + \alpha(T'(n'))$, which concludes that R is a successful natural reduction. □

3 Barriers for Proving Soundness of 2-Message SPS-ZK

In this section we prove our main result.

Theorem 3.1 (Barriers for Proving Soundness of 2-Message Arguments)**.** *Let $T(n)$ be a super-polynomial monotonically increasing function and assume the existence of $T(n)$-hard PRGs. Then, there exists an* **NP***-language L such that if (P, V) is a 2-message $T(n)$-simulatable argument for L with an efficient prover, then the following holds.*

1. *Consider some challenge $(C, t(\cdot))$ where C is PPT, and assume the existence of a natural reduction for basing soundness of (P, V) on* poly*-hardness of $(C, t(\cdot))$. Then $(C, t(\cdot))$ can be broken in PPT with advantage $1/\text{poly}(n')$ for all sufficiently large security parameters n'.*

2. *Consider some challenge $(C, t(\cdot))$ where C runs in probabilistic $\text{poly}(T(\cdot))$ time, and assume, further, that (P, V) is* strong *$T(\cdot)$-simulatable. Assume the existence of a natural reduction with security parameter blow-up c for basing soundness of (P, V) on $T'(\cdot)$-hardness of $(C, t(\cdot))$, where $T'(n) = T(n^{1/c})$. Then $(C, t(\cdot))$ can be broken in probabilistic $\text{poly}(T'(n'))$ time with advantage $1/\text{poly}(T'(n'))$ for all sufficiently large security parameters n'.*

By the result of [HILL99], the existence of one-way functions secure against $\text{poly}(T(n))$-size circuits implies the existence of PRGs secure against $\text{poly}(T(n))$-size circuits.[5]

Extensions to Nonuniform Reductions. Because Theorem 3.1 is proved for deterministic adversaries, by Lemma 2.7 we immediately also rule weakly nonuniform natural reduction. We also remark that we can deal with an even stronger form of a "fully nonuniform" reduction if we restrict to argument systems satisfying the stronger notion of *adaptive* (as opposed to non-adaptive) soundness; we present these results in Appendix B; see Theorem B.4. It remains an interesting open question to get the best of both and rule out (fully) nonuniform reductions even for the case of non-adaptive soundness.

Proof Outline. Following the "meta-reduction" paradigm by Boneh and Venkatesan [BV98] (which is also used in [Pas11, GW11, Pas12]), we will use R to directly break $(C, t(\cdot))$ with non-negligible probability. More formally, we exhibit a particular (inefficient) attacker A that breaks soundness of (P, V) with overwhelming probability, and we next show how to "emulate" this attacker for R efficiently without disturbing R's interaction with C. In particular, the proof follows the "two adversary" technique [Pas11, GW11, Wic13] in which, we construct two adversaries $A, \widetilde{A}$ where (1) A is inefficient, but is a successful attack against the soundness of the 2-message protocol. (2) $\widetilde{A}$

[5] Even though [HILL99] proved their result for $T(n) = \text{poly}(n)$, since it is black-box, it immediately "scales up" to handle larger $T(\cdot)$ as well.

is efficient and is *indistinguishable* from A in eyes of any efficient distinguisher. The first property implies that A can help the reduction win against the challenge $(C, t(\cdot))$. The second property means that once we switch from A to $\widetilde{A}$, the reduction (combined with the challenger C) will not "notice" this change (as their combined algorithm still constitutes and "efficient distinguisher"). Therefore, the efficient adversary R^A shall break the challenge, leading to the conclusion of the theorem. The actual proof goes through many steps and carefully designs an intermediate adversary $\widehat{A}$. See below for more details.

Proof of Theorem 3.1. We first prove the theorem for the "plain simulatability" of Case 1, and we then extend this proof to cover the "strong simulatability" of Case 2 as well.

Let $g : \{0,1\}^* \to \{0,1\}^*$ be a length-doubling non-uniformly hard PRG. Consider the language $L = \{g(s) \mid s \in \{0,1\}^*\}$ with witness relation $R_L(x) = \{s \in \{0,1\}^* \mid g(s) = x\}$.

Suppose (P, V) is a 2-message $T(\cdot)$-simulatable protocol for L, and P runs in polynomial time given any witness $w \in R_L(x)$. Suppose further that there exists an inverse polynomial α and a polynomial-time natural reduction R with security parameter blow-up c, such that R^A that breaks the assumption $(C, t(\cdot))$ with advantage $\alpha(n')$ given security parameter $1^{n'}$ whenever A is a deterministic (computationally unbounded) adversary that breaks soundness of (P, V) with probability $1/2$ on security parameter n (namely, $|x| = n$ where x is the statement of the proof system); since the security parameter blow-up is c, we have that $n' \leq n^c$.

Inefficient Attacker A. We first describe our (inefficient) attacker A, and next explain how to emulate it efficiently. More precisely (as in [Pas11]), we define a class of *deterministic* attackers A^f, parameterized by a function $f\colon \{0,1\}^* \to \{0,1\}^\infty$. Given that A^f is deterministic, we may assume without loss of generality that R never asks its oracle the same query twice.

Let $S = S(x, z)$ be the $T(n)$-time simulator for the verifier $V^*(x, z)$, who sends its auxiliary input z to the prover P to get a response a, and then simply outputs a. On input 1^n, A^f samples $x \leftarrow \{0,1\}^n$ using $f(1^n)$ as randomness, and then outputs x. Next, on input a "first message" q, $A^f(1^n)$ computes $a = S(x, q)$ using $f(x, q)$ as randomness, and responds with the message a.

In the following, let $\mathbf{RO} : \{0,1\}^* \to \{0,1\}^\infty$ be a uniformly distributed random oracle.

Two More Adversaries. First consider an alternative (still inefficient) attacker $\widehat{A}^f$ that selects $s \in \{0,1\}^{n/2}$ (again using $f(1^n)$ as the randomness), lets $x = g(s)$, and then proceeds just as A^f does. We also construct a probabilistic *polynomial-time* attacker $\widetilde{A}$ that emulates $A^{\mathbf{RO}}$. $\widetilde{A}(1^n)$ uniformly samples $s \in \{0,1\}^{n/2}$ and outputs $x = g(s)$; next, on input a first message q, $\widetilde{A}$ runs the honest prover strategy $P(x, s)$ on input the message q and outputs whatever P outputs.

Claim 3.2. *There exists a negligible function ν such that $A^{\mathbf{RO}}$ breaks soundness of (P, V) with probability $1 - \nu(n)$. Moreover, $R^{A^{\mathbf{RO}}}(1^{n'})$ breaks $(C, t(\cdot))$ with advantage $\alpha(n')/2$ for sufficiently large n'.*

Proof. First, note that by perfect completeness of the scheme, $\widetilde{A}$ convinces the verifier with probability 1. By $T(\cdot)$-simulatability, it follows that $\widehat{A}^f$ also convinces the verifier with probability $1 - \mathrm{negl}(n)$. Additionally, by the hardness of the PRG g, we have that $A^{\mathbf{RO}}$ also convinces the verifier with probability $1 - \mathrm{negl}(n)$. Furthermore, except with probability $2^{-n/2}$ (over the choice of $\mathbf{RO}$), $A^{\mathbf{RO}}$ selects a false statement $x \notin L$. Therefore, by a union bound, there exists some

negligible function ν such that with probability $1-\nu(n)$, $A^{\mathbf{RO}}$ selects a statement $x \notin L$ and convinces the verifier as well. This finishes the proof of the first part.

probability at least $1-2\nu(n)$ over the choice of a random oracle $f \leftarrow \mathbf{RO}$, A^f To prove the second part, note that by an averaging argument, with breaks soundness of (P,V) with probability at least $1/2$. We refer to any such f as being "good". For every good f, we have that $R^{A^f}(1^{n'})$ runs in time $\text{poly}(n')$ and breaks $(C,t(\cdot))$ with advantage $\alpha(n')$. Since by assumption $n'^{1/c} \leq n$, we have that $2\nu(n) \leq 2\nu(n'^{1/c} = \text{negl}(n')$ and thus by a union bound, it follows that $R^{A^{\mathbf{RO}}}(1^{n'})$ breaks $(C,t(\cdot))$ with advantage $\alpha(n')/2$ for sufficiently large n'. □

The following claim concludes the proof of the first part of Theorem 3.1 by letting $B = R^{\widetilde{A}}$.

Claim 3.3. *For sufficiently large n', $R^{\widetilde{A}}(1^{n'})$ breaks $(C,t(\cdot))$ with advantage at least $\alpha(n')/8$.*

Proof. By Claim 3.2, we have that $R^{A^{\mathbf{RO}}}(1^{n'})$ breaks $(C,t(\cdot))$ with advantage $\alpha(n')/2 \geq 1/\text{poly}(n')$ for sufficiently large n'. Recall the alternative efficient attacker $\widehat{A}$ defined above. The only difference between $A^{\mathbf{RO}}$ and $\widehat{A}^{\mathbf{RO}}$ is that the former samples a statement from U_n while the latter samples a statement from $g(U_{n/2})$. Recall that $R(1^{n'})$ only queries its oracle on the security parameter n. Now consider the combination of C, R and the executions of S as a single distinguisher D who wants to distinguish U_n from $g(U_{n/2})$. Such a distinguisher runs in time $\text{poly}(T(n))$, so by the $\text{poly}(T(n))$-indistinguishability of U_n and $g(U_{n/2})$, it follows that $R^{\widehat{A}^{\mathbf{RO}}}(1^n)$ breaks $(C,t(\cdot))$ with advantage $\alpha(n')/2 - \text{negl}(n) \geq \alpha(n')/4$ for sufficiently large n'.

Recall that R, without loss of generality, never asks the same query twice because we only consider deterministic adversaries. Therefore, the only difference between $\widehat{A}^{\mathbf{RO}}$ and $\widetilde{A}$ is that the former uses simulated proofs (of true statements) whereas the latter uses honestly generated proofs. Thus, intuitively, the claim should directly follow by the indistinguishability property of the simulation (and the fact that C and R are polynomial-size). There is a small catch: note that R can query its oracle on several first messages q, similar to the execution of a verifier V^* in a sequential composition of (P,V) (on the same fixed statement x). Indeed, by the same argument as in the sequential composition theorem for SPS simulation [Pas03], we will show that indistinguishability still holds. More precisely, let $m(n')$ be an upper-bound on the running-time of $R(1^{n'})$ (in this case, $m(n') = \text{poly}(n') = \text{poly}(n)$), and define a sequence of $m(n')$ hybrids $H_0, \ldots, H_{m(n')}$ as follows. The hybrid H_i is the output of C when interacting with $R^{(\cdot)}$ where the first i oracle responses (apart from the returned x) are simulated (i.e., answered by $\widehat{A}^{\mathbf{RO}}$), and the remaining queries are answered by running the honest prover strategy (i.e., answered by $\widetilde{A}$). Note that H_0 is the output of C after interacting with $R^{\widetilde{A}}(1^{n'})$, and $H_{m(n')}$ is the output of C after interacting with $R^{\widehat{A}^{\mathbf{RO}}}(1^{n'})$.

Indistinguishability of any two consecutive hybrids H_i and H_{i+1} follows by the indistinguishability of the simulation and the fact that oracle responses for all $j > i+1$ can be generated in polynomial-time (given the witness to the selected statement). More formally, if the outputs of hybrids H_i and H_{i+1} are $\frac{\alpha(n')}{8m(n')}$-distinguishable, we can always fix the first $i+1$ queries and the first i oracle responses so that the same $\frac{\alpha(n')}{8m(n')}$-distinguishability holds, and then use this fact to distinguish between an honest proof and a simulated proof (i.e., the answers to the $(i+1)^{th}$ query) with advantage $\frac{\alpha(n')}{8m(n')}$ (by answering the subsequent oracle queries efficiently using a hard-wired witness). But, since $n' \leq n^c$, we have that the distinguishing advantage $\frac{\alpha(n')}{8m(n')} \geq \frac{\alpha(n^c)}{8m(n^c)}$ (because α is an inverse polynomial); this contradicts the (nonuniform) indistinguishability of the simulation

from the honest proof of statements of length n. Thus, the statistical distance between the output bit of the challenger C in hybrids H_0 and $H_{m(n')}$ is at most $\frac{\alpha(n')}{8}$ for sufficiently large n'. Since $R^{\widehat{A}^{\mathbf{RO}}}(1^n)$ breaks $(C, t(\cdot))$ with advantage $\frac{\alpha(n')}{4}$ for sufficiently large n', the claim follows. □

Second Part of Theorem 3.1. We finally note that if (P, V) is *strong* $T(n)$-simulatable, then roughly the same argument as above works even if C and R are allowed to run in time $\mathrm{poly}(T'(n'))$. Roughly speaking, $\mathrm{negl}(n)$ probabilities/advantages would be replaced by $\mathrm{negl}(T(n))$, and $\alpha(n')$ will be replaced by $\alpha(T'(n'))$, but doing so requires leveraging strong $T(n)$-simulatability as well as dealing with some other subtleties.

Before continuing, we make some important observations about the running time of the reduction and the advantage α of breaking $(C, t(\cdot))$. By definition, reduction R runs in time $\mathrm{poly}(T'(n'))$, and given any adversary who breaks the soundness on message length n, R will win against C with advantage at least $\alpha(T'(n'))$ for some inverse polynomial $\alpha(\cdot)$. Because ${n'}^{1/c} \leq n$ and $T'(n) = T(n^{1/c})$, it holds that the reduction's running time and its (desired) advantage to win in challenge $(C, t(\cdot))$ satisfy the following bounds:

$$\mathrm{poly}(T'(n')) = \mathrm{poly}(T({n'}^{1/c})) \leq \mathrm{poly}(T(n)), \tag{1}$$

$$\alpha(T'(n')) = 1/\mathrm{poly}(T'(n')) \geq 1/\mathrm{poly}(T(n)), \tag{2}$$

where Equation 2 follows from Equation 1.

The proof now proceeds by showing the appropriate variants of Claim 3.2 and Claim 3.3.

Claim 3.4 (Variant of Claim 3.2)**.** *There is a negligible ν such that $A^{\mathbf{RO}}$ breaks soundness of (P, V) with probability $1 - \nu(T(n))$. Moreover, $R^{A^{\mathbf{RO}}}(1^{n'})$ breaks $(C, t(\cdot))$ with advantage $\alpha(T'(n'))/2$ for sufficiently large n'.*

Proof. Claim 3.4 can be proved similarly to Claim 3.2, while relying on strong $T(n)$-simulatability, perfect completeness and some additional crucial observations. For clarity, we repeat the the proof to highlight these subtleties. By perfect completeness of the scheme, $\widetilde{A}$ convinces the verifier with probability 1. By $T(\cdot)$-simulatability, $\widehat{A}^f$ also convinces the verifier with probability $1 - \mathrm{negl}(T(n))$. By $\mathrm{poly}(T(n))$ hardness of the PRG g, $A^{\mathbf{RO}}$ thus convinces the verifier with probability $1 - \mathrm{negl}(T(n))$. Furthermore, except with probability $2^{-n/2}$ (over the choice of $\mathbf{RO}$), $A^{\mathbf{RO}}$ selects a false statement $x \notin L$. We observe that $2^{-n/2} < \mathrm{negl}(T(n))$, since we assume the existence of $\mathrm{poly}(T(n))$-secure PRGs, and that implies $T(n) \in 2^{o(n)}$. We can therefore conclude, by a union bound, that there exists some negligible function ν such that with probability $1 - \nu(T(n))$, $A^{\mathbf{RO}}$ selects a statement $x \notin L$ and convinces the verifier as well. (Note that perfect completeness, or at least completeness with probability $1 - \mathrm{negl}(T(n))$ as opposed to just $\mathrm{negl}(n)$, is here being crucially used. See Remark 3.6 for more discussion on how to relax this condition.)

The second part then follows using the same averaging argument and a union bound as in the proof of Claim 3.2: By an averaging argument, with probability at least $1 - 2\nu(T(n))$ over the choice of a random oracle $f \leftarrow \mathbf{RO}$, A^f breaks soundness of (P, V) with probability at least $1/2$. Call any such f "good". By Equation 1, we have that $\mathrm{poly}(T(n)) \geq \mathrm{poly}(T'(n'))$, thus $2\nu(T(n) \leq 2\nu(\mathrm{poly}(T'(n'))) = \mathrm{negl}(T'(n'))$; thus the probability that f is good is at least $1 - \mathrm{negl}(T'(n'))$. Furthermore, note that for every good f, $R^{A^f}(1^{n'})$ runs in time $\mathrm{poly}(T'(n'))$ and breaks $(C, t(\cdot))$ with advantage $\alpha(T'(n'))$. By a union bound, it follows that $R^{A^{\mathbf{RO}}}(1^{n'})$ breaks $(C, t(\cdot))$ with advantage $\alpha(T'(n'))/2$ for sufficiently large n'. □

We next show the following variant of Claim 3.3.

Claim 3.5 (Variant of Claim 3.3)**.** *For sufficiently large n', $R^{\widetilde{A}}$ breaks $(C, t(\cdot))$ with advantage at least $\alpha(T'(n'))/8$.*

Proof. By Claim 3.4, we have that $R^{A^{\mathbf{RO}}}(1^{n'})$ breaks $(C, t(\cdot))$ with advantage $\alpha(T'(n'))/2$ for sufficiently large n'. Recall the alternative (inefficient) attacker $\widehat{A}$. The reduction can now run in time $\mathrm{poly}(T'(n')) \leq \mathrm{poly}(T(n))$ (by Equation 1), but again, by the $\mathrm{poly}(T(n))$-indistinguishability of U_n and $g(U_{n/2})$, it follows that $R^{\widehat{A}^{\mathbf{RO}}}(1^{n'})$ breaks $(C, t(\cdot))$ with advantage at least $\alpha(T(n'))/4$ for sufficiently large n. The reason is that otherwise, we can use C, R and the simulator S (who all run in time $\mathrm{poly}(T'(n')) \leq \mathrm{poly}(T(n))$ and break the $\mathrm{poly}(T(n))$-hardness of the PRG.

We now argue why the adversary $\widetilde{A}$ can be used instead of $\widehat{A}^{\mathbf{RO}}$ and derive Claim 3.5. We follow the same steps as in Claim 3.3. However, we shall use $m(n') = \mathrm{poly}(T'(n')) \leq \mathrm{poly}(T(n))$ hybrids, because the reduction R can call its oracle $\mathrm{poly}(T'(n'))$ times. Now, for every pair of consecutive hybrids H_i and H_{i+1} the distinguishability gap that could be obtained by any $\mathrm{poly}(T(n))$-time distinguisher is at most $\mathrm{negl}(T(n))$, due to the strong $T(n)$-simulatability property. Therefore, the statistical distance between the output bit of the challenger in hybrids H_0 and $H_{m(n')}$ is at most $m(n') \cdot \mathrm{negl}(T(n)) = \mathrm{negl}(T(n))$. Since $R^{\widehat{A}^{\mathbf{RO}}}(1^n)$ breaks $(C, t(\cdot))$ with advantage $\frac{\alpha(T'(n'))}{4}$ for sufficiently large n', it follows that $R^{\widetilde{A}}$ breaks $(C, t(\cdot))$ with advantage at least $\alpha(T'(n'))/4 - \mathrm{negl}(T(n))$. But, since $\alpha(T'(n')) \geq \frac{1}{\mathrm{poly}(T(n))}$ (by Equation 2) is non-negligible in $T(n)$, we have that $\alpha(T'(n'))/4 - \mathrm{negl}(T(n)) \geq \alpha(T'(n'))/8$, for sufficiently large n', which concludes the proof. □

□

We now briefly discuss some extensions to Theorem 3.1

Remark 3.6 (Completeness Error vs. Probability of Breaking the Challenge)**.** As stated in the proof of Claim 3.4, when we deal with strong super-polynomial simulation, we are crucially relying on perfect completeness. We here note relaxations that still enable the proof to go through.

Completeness Error $\mathrm{negl}(T(n))$**.** If we assume the completeness error is $\mathrm{negl}(T(n))$, (as opposed to the standard notion of completeness error $\mathrm{negl}(n)$), then the same proof still goes through, as noted already in the proof of Claim 3.4.

Intractability Assumption Advantage $1/\mathrm{poly}(n')$**.** If considering reductions that break the intractability assumption $(C, t(\cdot))$ with advantage $1/\mathrm{poly}(n')$ (rather than $1/\mathrm{poly}(T'(n'))$), then it suffices to assume completeness error $\mathrm{negl}(n)$. The reason for this is that $1/\mathrm{poly}(n') \geq 1/\mathrm{poly}(n)$ since the reduction is natural.

Threshold $t(n') = 0$**.** Finally, if the threshold $t(\cdot)$ in the challenge $(C, t(\cdot))$ is zero $t(n') = 0$, then we can again handle completeness error $\mathrm{negl}(n')$ (even when the challenge only needs to be broken with probability $1/\mathrm{poly}(T'(n'))$). The reason is that there is no way to win a challenge game with threshold $t(\cdot)$ with *negative* advantage (while doing so *is* possible when $t(n') > 0$) and hence we can obtain in the first part of Claim 3.4 that with probability $1 - \mathrm{negl}(n') - \mathrm{negl}(T(n)) = 1 - \mathrm{negl}(n')$ over $f \leftarrow \mathbf{RO}$, the adversary $A^{\mathbf{RO}}$ is successful in breaking soundness with probability $1/2$. Then, for any such good f, $R^{A^{\mathbf{RO}}}$ would break the challenge with advantage $\alpha(T'(n'))/2$ and for any bad f, $R^{A^{\mathbf{RO}}}$ still has advantage at least zero. Combining these two, $R^{A^{\mathbf{RO}}}$ still breaks the challenge with advantage $\alpha(T'(n'))/2$.

All $t(\cdot)$-trivial Assumptions. One can generalize the argument above for the case of $t(n') = 0$, to any $t(n')$ as long as winning the challenge $(C, t(\cdot))$ with advantage 0 (i.e., with probability $t(n')$) is possible (perhaps trivially) in $\text{poly}(T'(n'))$ time. This is of course the case of $t(n') = 0$, but more generally for natural scenarios, such as indistinguishability games in cryptography where $t(n') = 1/2$, this is possible by simply outputting a random bit. Formally, we refer to such assumptions as $t(\cdot)$-*trivial* assumptions. In that case, we need to minimally change the proof to make sure we never end up with a "too negative" advantage when using an adversary A. In particular, let A be the fixed adversary on security parameter n. Then, the reduction $R^A(1^{n'})$ can first "test" its adversary to make sure that $R^A(1^{n'})$ wins the challenge with advantage at least $-\alpha(n')/10$. This can be done by running $R^A(1^{n'})$ $\text{poly}(1/\alpha(n')) = \text{poly}(T'(n'))$ many times against the $\text{poly}(T'(n'))$-time adversary, and then rejecting A, if the average probability of success is below $t(n') - \alpha(n')/100$. If an adversary A is a successful adversary, (except with $\text{negl}(T'(n'))$ probability) $R^A(1^{n'})$ will pass this test. On the other hand, if A does not pass this test, we will simply use the trivial adversary who wins the challenge with non-negative advantage. Finally, if $R^A(1^{n'})$ wins the challenge with probability at most $t(n') - \alpha(n')/100$, (except with $\text{negl}(T'(n'))$ probability) A will fail the test. Putting these together, we obtain that the averaging argument of Claim 3.2 still holds as desired (with just a worse constant next to $\alpha(n')$).

References

[AGGM06] Adi Akavia, Oded Goldreich, Shafi Goldwasser, and Dana Moshkovitz. On basing one-way functions on NP-hardness. In *STOC '06*, pages 701–710, 2006. 4

[BCC88] Gilles Brassard, David Chaum, and Claude Crépeau. Minimum disclosure proofs of knowledge. *J. Comput. Syst. Sci.*, 37(2):156–189, 1988. 5

[BDSG+13] Nir Bitansky, Dana Dachman-Soled, Sanjam Garg, Abhishek Jain, Yael Tauman Kalai, Adriana López-Alt, and Daniel Wichs. Why "fiat-shamir for proofs" lacks a proof. In *Theory of cryptography conference*, pages 182–201. Springer, 2013. 3

[Blu86] M. Blum. How to prove a theorem so no one else can claim it. *Proc. of the International Congress of Mathematicians*, pages 1444–1451, 1986. 2, 3

[BMG07] Boaz Barak and Mohammad Mahmoody-Ghidary. Lower bounds on signatures from symmetric primitives. In *IEEE Symposium on Foundations of Computer Science (FOCS)*, 2007. 4

[Bra83] Gilles Brassard. Relativized cryptography. *IEEE Transactions on Information Theory*, 29(6):877–893, 1983. 4

[BT03] Andrej Bogdanov and Luca Trevisan. On worst-case to average-case reductions for np problems. In *FOCS*, pages 308–317, 2003. 4

[BV98] Dan Boneh and Ramarathnam Venkatesan. Breaking rsa may not be equivalent to factoring. In *EUROCRYPT*, pages 59–71, 1998. 9

[CCH+19] Ran Canetti, Yilei Chen, Justin Holmgren, Alex Lombardi, Guy N Rothblum, Ron D Rothblum, and Daniel Wichs. Fiat-shamir: from practice to theory. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1082–1090, 2019. 4

[CGGM00] Ran Canetti, Oded Goldreich, Shafi Goldwasser, and Silvio Micali. Resettable zero-knowledge (extended abstract). In *STOC '00*, pages 235–244, 2000. 2, 3

[CLMP13] Kai-Min Chung, Huijia Lin, Mohammad Mahmoody, and Rafael Pass. On the power of nonuniformity in proofs of security. *Innovations in Theoretical Computer Science*, 2013. 20, 21

[CLW18] Ran Canetti, Alex Lombardi, and Daniel Wichs. Fiat-shamir: from practice to theory, part ii (nizk and correlation intractability from circular-secure fhe). Technical report, Cryptology ePrint Archive: Report 2018/1248, 2018. 4

[DN07] Dwork and Naor. Zaps and their applications. *SICOMP: SIAM Journal on Computing*, 36, 2007. 3

[DOP05] Yevgeniy Dodis, Roberto Oliveira, and Krzysztof Pietrzak. On the generic insecurity of the full domain hash. In *CRYPTO*, pages 449–466, 2005. 6

[DSJKLA12] Dana Dachman-Soled, Abhishek Jain, Yael Tauman Kalai, and Adriana Lopez-Alt. On the (in)security of the fiat-shamir paradigm, revisited. *IACR Cryptology ePrint Archive*, 2012, 2012. 3

[FF93] Joan Feigenbaum and Lance Fortnow. Random-self-reducibility of complete sets. *SIAM Journal on Computing*, 22(5):994–1005, 1993. 4

[FS90] Uriel Feige and Adi Shamir. Witness indistinguishable and witness hiding protocols. In *STOC '90*, pages 416–426, 1990. 2

[GK96] Oded Goldreich and Ariel Kahan. How to construct constant-round zero-knowledge proof systems for NP. *Journal of Cryptology*, 9(3):167–190, 1996. 2

[GKM+00] Yael Gertner, Sampath Kannan, Tal Malkin, Omer Reingold, and Mahesh Viswanathan. The Relationship between Public Key Encryption and Oblivious transfer. In *FOCS*, pages 325–335, 2000. 4

[GMR89] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. The knowledge complexity of interactive proof systems. *SIAM Journal on Computing*, 18(1):186–208, 1989. 5

[GO94] Oded Goldreich and Yair Oren. Definitions and properties of zero-knowledge proof systems. *Journal of Cryptology*, 7:1–32, 1994. 2

[GW11] Craig Gentry and Daniel Wichs. Separating succinct non-interactive arguments from all falsifiable assumptions. In *STOC*, pages 99–108, 2011. 4, 6, 9

[HH09] Iftach Haitner and Thomas Holenstein. On the (im)possibility of key dependent encryption. In *TCC*, pages 202–219, 2009. 6

[HHRS07] Iftach Haitner, Jonathan J. Hoch, Omer Reingold, and Gil Segev. Finding collisions in interactive protocols – A tight lower bound on the round complexity of statistically-hiding commitments. In *Proceedings of the 47th Annual Symposium on Foundations of Computer Science (FOCS)*. IEEE Computer Society, 2007. 4

[HILL99] Johan Håstad, Russell Impagliazzo, Leonid Levin, and Michael Luby. A pseudorandom generator from any one-way function. *SIAM Journal on Computing*, 28:12–24, 1999. 9

[IR88] Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *CRYPTO '88*, pages 8–26, 1988. 4

[KKS18] Yael Tauman Kalai, Dakshita Khurana, and Amit Sahai. Statistical witness indistinguishability (and more) in two messages. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 34–65. Springer, 2018. 3

[KS17] Dakshita Khurana and Amit Sahai. How to achieve non-malleability in one or two rounds. In *2017 IEEE 58th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 564–575. IEEE, 2017. 3

[Nao03] Moni Naor. On cryptographic assumptions and challenges. In *CRYPTO*, pages 96–109, 2003. 6

[Pas03] Rafael Pass. Simulation in quasi-polynomial time, and its application to protocol composition. In *EUROCRYPT*, pages 160–176, 2003. 2, 3, 6, 11

[Pas06] Rafael Pass. Parallel repetition of zero-knowledge proofs and the possibility of basing cryptography on np-hardness. In *IEEE Conference on Computational Complexity*, pages 96–110, 2006. 4

[Pas11] Rafael Pass. Limits of provable security from standard assumptions. In *STOC*, pages 109–118, 2011. 4, 6, 9, 10

[Pas12] Rafael Pass. Barriers to provable non-interactive zero-knowledge and non-malleable commitments. Manuscript, 2012. 5, 9

[PS19] Chris Peikert and Sina Shiehian. Noninteractive zero knowledge for np from (plain) learning with errors. In *Annual International Cryptology Conference*, pages 89–114. Springer, 2019. 4

[PTV11] Rafael Pass, Wei-Lung Dustin Tseng, and Muthuramakrishnan Venkitasubramaniam. Towards non-black-box lower bounds in cryptography. In *TCC*, pages 579–596, 2011. 4

[RTV04] Omer Reingold, Luca Trevisan, and Salil P. Vadhan. Notions of reducibility between cryptographic primitives. In *TCC*, pages 1–20, 2004. 4

[RV10] Guy N. Rothblum and Salil P. Vadhan. Are pcps inherent in efficient arguments? *Computational Complexity*, 19(2):265–304, 2010. 6

[Sim98] Daniel R. Simon. Finding Collisions on a One-Way Street: Can Secure Hash Functions Be Based on General Assumptions? In *EUROCRYPT*, pages 334–345, 1998. 4

[Unr07] Dominique Unruh. Random oracles and auxiliary input. In Alfred Menezes, editor, *Advances in Cryptology - CRYPTO 2007, 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007, Proceedings*, volume 4622 of *Lecture Notes in Computer Science*, pages 205–223. Springer, 2007. 21

[Wic13] Daniel Wichs. Barriers in cryptography with weak, correlated and leaky sources. In *Proceedings of the 4th conference on Innovations in Theoretical Computer Science*, pages 111–126, 2013. 9

A Natural vs. Semi-natural Reductions

In this section, we further discuss the choices made in Definition 2.5 and justify them. In particular, we show that a less restrictive notion of a reduction implies the existence of a reduction satisfying the restrictions of Definition 2.5. We start by defining a less restrictive form a reduction, which allows the reduction to query the attacker on multiple security parameters. In particular, the notion of *semi-natural reduction* is defined exactly as that of a natural reduction, except that we skip condition (2).

Definition A.1 (Semi-natural Reductions)**.** We say that R is a *semi-natural* black-box reduction to breaking $\mathcal{C} = (C, t(\cdot))$, if R defined as in Definition 2.4, except we now *only* require the first condition to the satisfied.

Note that in contrast to a natural reduction, a semi-natural reduction is allowed to call its adversary on *multiple* security parameters $n_1, n_2, \ldots$. The key result in this section is to show that semi-natural reductions can be turned into natural ones. In fact, we shall show something stronger: It will suffice to have a semi-natural reduction that only succeeds "infinitely-often" to get a successful natural reduction. Towards formalizing this, we first formally provide a general definition of what it means for a reduction to be successful.[6]

Breaking a Security Game. In the rest of the section, we will consider some *fixed* security game $\mathcal{C} = (C, t(\cdot))$, and we study reductions from breaking $\mathcal{C}$ to breaking another fixed security-game $\mathcal{P} = (P, s(\cdot))$. For both of them, we will also fix some *concrete success probability advantages* $\varepsilon(\cdot), \delta(\cdot)$. For a security parameter n, we say that an adversary A_n *breaks* $\mathcal{P}$, if it breaks it with advantage at least $\varepsilon(n)$ on security parameter n; for a security parameter n', we say that the reduction R *breaks* $\mathcal{C}$, if it breaks it with advantage at least $\delta(n')$ on security parameter n'. Note that Definition 2.5 can be viewed as a special case where $\varepsilon(n) = 1/2$ and $\delta(\cdot)$ is some a-priori fixed inverse polynomial (i.e., the success probability of the reduction when having access to an attacker succeeding with probability $1/2$.)

To deal with more general primitives (e.g., digital signatures secure under adaptive chosen message attacks), where the attacker's communication complexity is not a-priori bounded, we will

[6] Note that in Definition 2.5, we only did this for the special security game of interest in this paper, that is, breaking soundness of a 2-round argument.

also allow both the reduction and the success probability to depend on the total communication length; towards this, we fix $\ell(n')$ as an upper bound on the total message lengths that the adversary sends to the reduction R when the reduction's security parameter is n'.

Defining Success of Security Reductions. We start by generalizing the notion of a successful natural reduction (i.e. generalizing Definition 2.5 to general security games). We here consider a generalization where the reduction only needs to succeed on some (pre-defined) infinite set of security parameters J. (Recall that we have here fixed some functions ε, δ, and the notion of breaking is defined with respect to those fixed bounds.)

Definition A.2 (Success Criteria for Natural Reductions)**.** Let R is be a natural reduction. We call R *successful* over security parameters J if the following holds. Consider $n' \in J$, and assume that $R(1^{n'})$ calls its oracle on security parameter n. Then, if A breaks $\mathcal{P}$ on the security parameter n, we have that R^A breaks $\mathcal{C}$ on security parameter n'.

For the case of semi-natural reductions, we will consider an even weaker notion of "infinitely-often" success: we only require the reduction to succeed on infinitely many security parameters n' if given access to any attacker that succeeds on infinitely many security parameters n. To keep this notion comparable with the success criteria for the natural setting, we again consider a generalization where the infinitely-often criteria needs to hold on any infinite subset of some (pre-defined) infinite set of security parameters I for the adversary.

Definition A.3 (Success Criteria for Semi-natural Reductions)**.** Let R be a semi-natural reduction from breaking $\mathcal{C}$ to breaking $\mathcal{P}$. We call R (infinitely-often) *successful* over a set of security parameters I if the following holds. For every infinite set $I' \subseteq I$, there is an infinite set J such that if A breaks $\mathcal{P}$ on all security parameters in I', then R^A breaks $\mathcal{C}$ on all security parameters in J.

Before proving the main result of this section, we observe that this infinitely-often notion of success indeed is weaker than the notion of success we used for natural reductions.

Proposition A.4. *If R is a successful natural reduction over an infinite set J, then it is also an (infinitely-often) successful semi-natural reduction over some infinite set I.*

Proof. If R is a successful natural reduction over an infinite set J, and let c be the associated security parameter blow-up. For every n', let $n = n(n')$ be the relevant security parameter for the adversary such that the reduction uses any successful adversaries over security parameter n to break the challenger $\mathcal{C}$ over security parameter n'. Define $I = \{n(n') \mid n' \in J\}$. By the security parameter blow-up property of a semi-natural reduction, recall that $n' \leq (n(n'))^c$. Therefore, I has to be infinite, since $n' \in J$ can be larger than any integer, and therefore $n(n') \geq n^{1/c}$ also has to be larger than any integer.

We claim that R is a successful semi-natural reduction over the set I. Let $I' \subseteq I$ be an infinite subset of I. Then, consider $J' = \{n' \mid n(n') \in I'\}$. By the definition of J' and the fact that R is a successful natural reduction, if an adversary succeeds on all security parameters in I', R^A breaks the challenge on all security parameters in J' as well. As a result, R is also an (infinitely-often) successful semi-natural reduction over I. □

We now prove the main result of this section, showing that semi-natural reductions can be turned into successful natural reductions.

Lemma A.5 (Sparsification Lemma: From Semi-natural to Natural Reductions)**.** *Let $T(\cdot)$ be a monotonically increasing function such that $T(n) < 2^n$, and let R be a $T(\cdot)$-time successful semi-natural reduction from breaking $\mathcal{P}$ to breaking $\mathcal{C}$ over the natural numbers $\mathbb{N}$. Then there is an infinite set J and successful natural reduction R' over J that runs in time* $\text{poly}(T(\cdot))$.

Proof. The idea is to first choose a sparse subset I' of adversary security parameters, such that a semi-natural reduction R will not be able to query A on *two different* security parameters in I'. We then define J to essentially be the set of security parameters for $\mathcal{C}$ that the reduction R wins on when given a successful adversary over security parameters $n \in I'$. The formal argument follows.

Suppose $R(1^{n'})$ runs in time $T(n') < 2^{n'}$ and has security blow-up c. Let $F(n) = 2^{n^c}$, and defined the set I' as $\{F(1), F(F(1)), F(F(F(1))), \cdots\}$. We claim that for any $n' \in N$, $R(1^{n'})$ can call A on at most one security parameter $n \in I'$. For sake of contradiction, suppose the reduction asks A on $n_1 < n_2 \in I'$. Then, by the security blow-up condition of R, we have

$$n' \leq n_1^c. \tag{3}$$

In addition, by definition of I', we have $2^{n_1^c} \leq n_2$, which together with Equation 3 implies that

$$2^{n'} \leq 2^{n_1^c} \leq n_2. \tag{4}$$

On the other hand, since R needs to provide the security parameter in unary, 1^n, to its attacker while running in time at most $T(n')$, it holds that

$$n_2 \leq T(n') < 2^{n'},$$

which contradicts Equation 4.

We now let $R'(1^{n'})$ be the reduction that runs $R(1^{n'})$ and only asks a query to A if it is for a security parameter $n \in I'$. Note that since R' needs to query its attacker on the security parameter n in unary, 1^n, it suffices to be able to decide whether $n \in I'$ in time $\text{poly}(2^n)$, to conclude that R' has polynomial overhead over R; to do this, we simply iterate F until we either hit n, or reach a number that exceeds 2^n. If $n \notin I'$, then R' simply internally emulates the response of A by answering $\perp$. Since by the above argument, $R(1^{n'})$ only queries its oracle on a single security parameter in I', we have that R' is a natural reduction. We now show that R' is also successful. Recall that R is an infinitely-often successful reduction. Since I' is infinite, there exists be an infinite set J (depending on I') such that, whenever A is a successful attack on I', then R^J succeeds in breaking $\mathcal{C}$ on security parameters in J. We now state and prove that R' is a successful natural reduction over J, which concludes the proof of Lemma A.5.

Claim A.6. *Consider some attacker A that is successful on the set $\mathbb{N}$. Then there exists some infinite set J such that R'^A succeeds in breaking $\mathcal{C}$ over security parameters in J.*

Proof. Let the adversary A' be defined as A on security parameters in I', and simply answering $\perp$ on all other security parameters. Recall that R' does not call its adversary on any security parameter outside I', so R'^A and $R'^{A'}$ behave exactly the same. Moreover, $R^{A'}$ succeeds on security parameters in J, by the definition of J and that the fact that A' is successful over security parameters in I'. Therefore, $R'^A = R'^A$ also succeeds in breaking $\mathcal{C}$ for security parameters in J. □

□

B Barriers for Nonuniform Reductions and Adaptive Soundness

Here we discuss extensions to our Theorem 3.1 to arbitrary nonuniform reductions, when the reductions job is to prove *adaptive* soundness of the 2-message scheme.

Definition B.1 (Interactive Arguments with Adaptive Soundness)**.** Let (P, V) be a 2-message interactive argument for the language L, where the verifier's first message is only a function of the length of the statement x to be proven. We say that an adversary P^* *breaks the adaptive soundness of* (P, V) *with probability* $\mu(n)$, if

$$\Pr\left[r \leftarrow \{0,1\}^\infty, q \leftarrow V_r(1^n); (x, \mathsf{aux}) \leftarrow P^*(1^n, q) : \langle P^*(\mathsf{aux}), V_r\rangle(x) = 1 \wedge x \notin L \cap \{0,1\}^n\right] \geq \mu(n)$$

where V_r denotes V with randomness fixed to r. We say that (P, V) satisfies *adaptive soundness*, if for any PPT adversary that breaks adaptive soundness of (P, V) with probability $\mu(n)$, it holds that $\mu(n) = \mathrm{negl}(n)$.

We now define a general definition for nonuniform natural reduction; this definition is adapted from [CLMP13]. We next define what it means for such a reduction to be successful in reducing adaptive soundness to the hardness of some security game.

Definition B.2 (Nonuniform Natural Reductions)**.** We say that R is a *nonuniform* natural black-box reduction to breaking $\mathcal{C} = (C, t(\cdot))$, if R is an oracle machine such that the following holds. For every security parameter n', there exists some fixed security parameter n such that for every z, $R(1^{n'}, z)$ only invokes its oracle on security parameter n. Moreover, there exists some constant c (referred to as the security parameter blow-up) such that for every such n', n, we have that $n' \leq n^c$.

Definition B.3 (Nonuniform Natural Reductions for Adaptive Soundness of 2-Message Arguments)**.** Let $(C, t(\cdot))$ be an assumption and (P, V) a 2-message argument with adaptive soundness syntax (for its verifier). We call R a *nonuniform* natural reduction for basing adaptive soundness of (P, V) on a $T'(\cdot)$-hardness of challenger $(C, t(\cdot))$, if R is a nonuniform natural reduction and there exists some inverse polynomial function α and a function $Z(\cdot)$ such that the following holds. Consider some n, n' such that $R^A(1^{n'})$ invokes its attacker on security parameter n, and let A be an arbitrary deterministic attacker that breaks the adaptive soundness of (P, V) with probability at least $1/2$ on security parameter (i.e., statement length) n. Then $R^A(1^{n'}, Z(A))$ runs in probabilistic time $\mathrm{poly}(T'(n'))$ and breaks $(C, t(\cdot))$ over security parameter n' with probability $t(n') + \alpha(T'(n'))$.

We proceed to state the extension of Theorem 3.1 to the nonuniform setting; note that we rule out a larger set of "fully" nonuniformnm reduction, but the impossibility result only applies to arguments satisfying the stronger notion of adaptive soundness.

Theorem B.4 (Barriers against Nonuniform Reductions for Adaptive Soundness)**.** *Let $T(n)$ be a super-polynomial monotonically increasing function and assume the existence of $T(n)$-hard PRGs. Then, there exists an* **NP***-language L such that if (P, V) is a 2-message $T(n)$-simulatable argument for L with an efficient prover, then the following holds.*

1. *Consider some challenge $(C, t(\cdot))$ where C is PPT, and assume the existence of a natural nonuniform reduction for basing adaptive soundness of (P, V) on* poly*-hardness of $(C, t(\cdot))$. Then $(C, t(\cdot))$ can be broken in nonuniform PPT with advantage $1/\mathrm{poly}(n')$ for all sufficiently large security parameters n'.*

2. *Consider some challenge* $(C, t(\cdot))$ *where* C *runs in probabilistic* $\text{poly}(T(\cdot))$ *time, and assume, further, that* (P, V) *is* strong $T(\cdot)$*-simulatable. Assume the existence of a natural nonuniform reduction with security parameter blow-up* c *for basing adaptive soundness of* (P, V) *on* $T'(\cdot)$*-hardness of* $(C, t(\cdot))$*, where* $T'(n) = T(n^{1/c})$*. Then* $(C, t(\cdot))$ *can be broken in nonuniform* $\text{poly}(T'(n'))$ *time with advantage* $1/\text{poly}(T'(n'))$ *for sufficiently large security parameters* n'*.*

Proof. At a high level, the proof is similar to that of Theorem 3.1, but we also use Lemma B.5 below, due to [Unr07], which was also previously used in [CLMP13] to prove separation for nonuniform security reductions.

Notation. Suppose $\mathcal{S}$ is a set of oracle queries and f is an oracle. By $f_{\mathcal{S}}$ we refer to the *partial* function that is defined over $\mathcal{S}$ and is equal to f on those points.

Lemma B.5 (Re-sampling of Random Oracles under Nonuniform Advice [Unr07])**.** *Suppose* D *is a (computationally unbounded) oracle algorithm that receives an nonuniform advice* z *of length* $|z| = d$ *about an oracle* $f \leftarrow \mathbf{RO}$ *and asks* u *queries to its oracle, and suppose* z *is computed using a function* Z *that maps* f *to* $\{0,1\}^d$*. Then for every integer* w*, there is an (inefficient) function* Samp *that gets as input some* $z \in \{0,1\}^d$ *and* f *and outputs a set* $\mathcal{S}$ *of* w *points in the domain of* f *such that the view of* D *in the following two experiments is* $\sqrt{du/(2w)}$ *statistically close:*

- (1) $f \leftarrow \mathbf{RO}$*,* (2) $z = z(f)$*, and* (3) *Execute* $D^f(z)$*.*
- (1) $f \leftarrow \mathbf{RO}$*,* (2) $z = z(f)$*,* (3)*, Get* $\mathcal{S} \leftarrow \mathsf{Samp}(z, f)$ (4) *Re-sample* $f' \leftarrow (\mathbf{RO}|f_{\mathcal{S}})$ *(i.e., sample* $f' \leftarrow \mathbf{RO}$ *conditioned on answers to* $\mathcal{S}$ *not changing), and* (5) *Execute* $D^{f'}(z)$*.*

Informally speaking, we use Lemma B.5 to fix part of the inefficient oracle (in the proof of Theorem 3.1) and re-sample the rest of the adversary at random, and this way we essentially make the nonuniform advice independent of the oracle. This allows us to switch the adversary from inefficient to efficient on *every other non-fixed point* without the reduction noticing it.

We will only write the proof for the first part of Theorem B.4, and the second part will be a straightforward adaptation of the proof of the second part of Theorem 3.1 using the same exact technique that we apply to the first part of Theorem B.4.

We use the same PRG g and same language L as in the proof of Theorem 3.1. Again assume (P, V) is a 2-message $T(\cdot)$-simulatable protocol for L with simulator S and in which P runs in polynomial time.

Suppose further that there exists an inverse polynomial α and a polynomial-time natural nonuniform reduction R with corresponding auxiliary-input selecting function Z and security parameter blow-up c, such that $R^A(1^{n'}, Z(A))$ that breaks the assumption $(C, t(\cdot))$ with advantage $\alpha(n')$ given security parameter $1^{n'}$ whenever A is a deterministic (computationally unbounded) adversary that breaks soundness of (P, V) with probability $1/2$ on security parameter n.

Let $S = S(x, z')$ be the $T(n)$-time simulator for the verifier $V^*(x, z')$, who sends its auxiliary input z' to the prover P to get a response a, and then simply outputs a. Consider some security parameter n' and let n denotes the security parameter $R(1^{n'}, \cdot)$ invokes its oracle on.

Adversaries. We define the following adversaries based on an oracle $f\colon \{0,1\}^* \mapsto \{0,1\}^\infty$.

- **Inefficient Adversary** A^f: Given query $(1^n, q)$, A^f uses $f(1^n, q)$ to sample (x, r) at random, where $x \in \{0,1\}^n$ is a statement and r is some randomness for S; A next computes $a = S(x, q)$ using r as randomness and outputs (x, a).

- **Inefficient Adversary** $\widehat{A}^f$: Given query $(1^n, q)$, $\widehat{A}^f$ uses $f(1^n, q)$ to sample (s, r) at random, where $s \in \{0,1\}^{n/2}$ and r is some randomness for S; $\widehat{A}$ next computes $x = g(s)$, and computes a (by running S and r) just as A does.
- **Efficient Adversary** $\widetilde{A}$: Given any new query $(1^n, q)$, $\widetilde{A}$ samples a random string $s \in \{0,1\}^{n/2}$, lets $x = g(s)$, and generates a honestly generated proof (using P) of x given the first message q and using witness s.

We now go over Claim 3.2 and Claim 3.3 and adapt their statements and proofs to the new nonuniform setting. Claim 3.2 holds exactly as stated. One difference in the proof is that the three adversaries have a different syntax, but they are still indistinguishable in eyes of a polynomial-time verifier, which is what we need to prove the first part of Claim 3.2. For the second part, by a similar union bound, it again follows that $R^{A^{\mathbf{RO}}}(1^n, Z(A^{\mathbf{RO}}))$ breaks $(C, t(\cdot))$ with advantage $\alpha(n')/2$ for sufficiently large n'.

Claim 3.3 needs to change substantially. We cannot simply switch between the different adversaries without the reduction noticing it, due to the nonuniform advice. What we will do, however, is to use Lemma B.5 as follows. We will *fix* the inefficient adversary's responses to some of the queries, so that the remaining part of the oracle **RO** is almost completely independent of the advice (at least as much as the reduction can notice it through its interaction with its oracle). We then hard-wire the answers into the code of the reduction R' that breaks the challenge, and use the efficient adversary to answer the remaining queries. The remaining steps are similar to the uniform setting.

More formally, below, we will define a *distribution* over non-uniform PPT attackers B (more precisely, a distribution over the nonuniform advice to be given to B) and show that B (with the sampled advice z) breaks (C, t) with inverse polynomial advantage. This directly yields the theorem as we can then simply nonuniformly fix the best nonuniform advice z.

We turn to defining the attacker B and the nonuniform advice it takes.

The Nonuniform Challenge Breaker B: B on input $1^{n'}$ proceeds as follows:

1. Sample $f \leftarrow \mathbf{RO}$ and interpret this as the oracle that defines the inefficient adversary A^f on security parameter n'.
2. Let $z = Z(A^f)$ be the nonuniform advice to be given to the security reduction R.
3. Interpret C communicating with R^A as D of Lemma B.5 and apply this lemma with the following parameters: Let $u = \text{poly}(n')$ be the query complexity of the reduction R and $d = |z|$. Pick $w = \text{poly}(n)$ large enough such that $\sqrt{d \cdot u/(2w)} \leq \alpha(n')/4$, and run Samp of Lemma B.5 to obtain the set $\mathcal{S}$.
4. For every query $c = (1^n, q) \in \mathcal{S}$, let a_c be the simulated answer generated by the simulator as executed by the adversary A^f on query c using randomness $f(1^n, q)$.
5. Emulate R^{A_f} but for any query $c \in \mathcal{S}$, provide the "hard-wired" answer a_c; for any other query $c' \notin \mathcal{S}$, use the efficient adversary $\widetilde{A}$ to answer c.

Before proving the claim, note that in the above description of the attacker B, the last emulation step—which is the only step that will communicate with the external challenger—can be performed

in PPT. This final step, however, requires the earlier steps to generate some nonuniform advice; this advice consists of (1) the polynomial-size advice z (used by R), and (2) "hard-coded answers" to a polynomial sized set of queries.

Let us now prove that this attacker B is successful.

Claim B.6. *$B(1^{n'})$ wins in the challenge $(C, t(\cdot))$ with advantage at least $\alpha(n')/16$ given security parameter n'.*

Proof. To analyze the success probability of B, consider a hybrid attacker B' that in the the last step, samples a new random oracle $f' \leftarrow (\mathbf{RO}|f_{\mathcal{S}})$, and emulates answers to queries $c' \notin \mathcal{S}$ by using the inefficient attacker $A^{f'}$. By Lemma B.5, it holds that the output of C in an interaction with B' given security parameter n' is $\alpha(n')/4$-statistically close to the output of C in an interaction $R^{A^{\mathbf{RO}}}$. Since by the (analog of) Claim 3.2, $R^{A^{\mathbf{RO}}}$ wins with advantage $\alpha(n')/2$, we have that B' wins with advantage at least $\alpha(n')/4$. However, note that now, the oracle f' is completely independent of the advice z given to the reduction B and the partially fixed oracle $f_{\mathcal{S}}$. Therefore, using exactly the same argument as in the proof of Claim 3.3, we can switch from emulating the last step using the inefficient adversary $A^{f'}$ to instead using the efficient adversary $\widetilde{A}$ and still preserve non-negligible probability $(\alpha(n')/4)/4 = \alpha(n')/16$ of winning in the challenge $(C, t(\cdot))$. □

□