Why "Fiat-Shamir for Proofs" Lacks a Proof*

Nir Bitansky[1,**], Dana Dachman-Soled[2], Sanjam Garg[3,***], Abhishek Jain[4,†], Yael Tauman Kalai[2], Adriana López-Alt[5,‡], and Daniel Wichs[6]

[1] Tel Aviv University
[2] Microsoft Research New England
[3] UCLA
[4] MIT and BU
[5] NYU
[6] IBM Research, T.J. Watson

Abstract. The Fiat-Shamir heuristic [CRYPTO '86] is used to convert any 3-message public-coin proof or argument system into a non-interactive argument, by hashing the prover's first message to select the verifier's challenge. It is known that this heuristic is sound when the hash function is modeled as a random oracle. On the other hand, the surprising result of Goldwasser and Kalai [FOCS '03] shows that there exists a computationally sound *argument* on which the Fiat-Shamir heuristic is *never* sound, when instantiated with any *actual* efficient hash function.

This leaves us with the following interesting possibility: perhaps we can securely instantiates the Fiat-Shamir heuristic for all 3-message public-coin *statistically sound proofs*, even if we must fail for some computationally sound arguments. Indeed, this has been conjectured to be the case by Barak, Lindell and Vadhan [FOCS '03], but we do not have any provably secure instantiation under any "standard assumption". In this work, we give a broad black-box separation result showing that the security of the Fiat-Shamir heuristic for statistically sound proofs cannot be proved under virtually any *standard assumption* via a *black-box reduction*. More precisely:

- If we want to have a "universal" instantiation of the Fiat-Shamir heuristic that works for *all* 3-message public-coin proofs, then we cannot prove its security via a black-box reduction from any assumption that has the format of a "cryptographic game".
- For many concrete proof systems, if we want to have a "specific" instantiation of the Fiat-Shamir heuristic for that proof system, then we cannot prove its security via a black box reduction from any "falsifiable assumption" that has the format of a cryptographic game with an efficient challenger.

* This is an abridged merge of [BGW12] and [DJKL12]. See ePrint for full versions.

** Research was done while visiting IBM T.J., Watson Research Center. Supported by the Check Point Institute for Information Security, an ISF grant 20006317, and the Fulbright program.

*** Research conducted while at the IBM Research, T.J.Watson funded by NSF Grant No.1017660.

† Research conducted while at Microsoft Research New England.

‡ Research conducted while at Microsoft Research New England.

A. Sahai (Ed.): TCC 2013, LNCS 7785, pp. 182–201, 2013.

1 Introduction

The Fiat-Shamir (FS) heuristic [FS86] allows us to convert an interactive *public-coin* protocol between a *prover* P and a *verifier* V into a one-message (non-interactive) protocol. Recall that, in a public-coin protocol, the verifier sends a uniformly random *challenge* to the prover in each round. Under the FS heuristic, the prover executes the original interactive protocol "in his head", computing the verifier's challenge in each round by applying some *public hash function* to the transcript of the protocol so far. The prover then only sends the final protocol transcript to the actual verifier, who verifies its validity. The hash function can be initialized with some randomly chosen public seed, which we think of as a "common random string (CRS)", and therefore the compiled protocol is non-interactive in the CRS model. Alternatively, the seed can also be chosen by the verifier in an additional initial message, in which case the compiled protocol consists of two messages. This heuristic has numerous remarkable applications in cryptography, such as constructing practical *signature schemes* [Sch91, GQ90, Oka93], *non-interactive zero knowledge (NIZK)* [BR93], and non-interactive succinct arguments [Mic00].

Soundness of FS. Although the FS heuristic seems to produce secure cryptographic schemes in practice, its formal security properties remain elusive. Perhaps the most basic question is to understand the *soundness* of the heuristic when applied to a statistically sound *proof* or computationally sound *argument* for some NP language. We say that an instance of the FS-heuristic is sound if the resulting non-interactive protocol is a computationally sound argument, for the same language. We can ask what kind of protocols do we need to start with, and what kind of hash functions should we use, to make the FS-heuristic sound. Since we are interested in a negative result, we restrict our attention to *3-message public-coin (3PC)* protocols.

Applying FS to Arguments. On the positive side, if the FS heuristic uses a *random oracle* as its hash function, then it is known to be sound when applied to *any* 3PC argument [BR93, PS00, AABN02]. On the other hand, the work of Goldwasser and Kalai [GK03] shows a surprising negative result: the FS heuristic *cannot* be securely instantiated with any *actual* efficient hash function that would achieve the same result. In particular, there exists *some* 3PC argument on which the FS heuristic is *never* sound, no matter which efficient hash function we try to instantiate it with.

Applying FS to Proofs. The above negative result only applies to computationally sound arguments, and therefore we are still left with the following interesting possibility: perhaps the FS heuristic could be instantiated with some hash function that makes it sound for *all* 3PC statistically sound *proofs*, even if it can fail for some arguments. We call such a hash function *FS-universal*. When instantiated with an FS-universal hash function, the FS heuristic should successfully compile any 3PC proof into a non-interactive (computationally sound) argument.

Barak, Lindell, and Vadhan [BLV03] conjecture that such FS-universal hash functions should indeed exist, and define a plausible hash-function property called *entropy-preservation*, which they show to be sufficient. Variants of this entropy-preservation property were further studied by Dodis, Ristenpart and Vadhan [DRV12], who also showed that it is *necessary*. Nevertheless, despite the amazing possibility that such hash functions may exist, we do not have any candidate construction that is provably secure under some "standard" cryptographic hardness assumption.

Less ambitiously, we may hope to securely instantiate the Fiat-Shamir heuristic for many specific 3PC proof and argument systems. In particular, for some candidate 3PC proof or argument Π, we can hope to have a *FS(Π)-secure hash function* that preserves soundness when applying the FS heuristic specifically to the protocol Π. We do not know how to construct such FS(Π)-secure hash functions for essentially any "interesting" proof or argument system Π.

1.1 Our Results

In this work, we re-examine the possibility of having *FS-universal* hash functions, or *FS(Π)-secure* hash functions for specific proof systems Π. We prove broad black-box separation results showing that the security of such hash functions cannot be proved under virtually any *standard assumption* via a *black-box reduction* that treats the attacker as a black box. More specifically, we provide two main results:

FS-Universal Hash Functions. We show that one cannot prove the security of an FS-universal hash function via a black-box reduction from any "cryptographic game assumption" (see below). We leverage the connection of [BLV03, DRV12] between FS-universal and entropy preserving hash functions. Specifically, we first provide a separation for entropy preserving hash functions, and then use it to get a similar separation for FS-universal hash functions.

FS(Π)-Secure Hash Functions. For many specific proof and argument systems Π, we show that one cannot prove the FS(Π)-security of a hash function via a black-box reduction from any "falsifiable assumption" (see below). In particular, we first prove a black-box impossibility result for two-round zero knowledge w.r.t. super-poynomial simulation, extending the result of Goldreich and Oren [GO94]. Then, by relying on this result, we obtain a black-box impossibility result for any proof/argument system Π for a sub-exponentially hard language $\mathcal{L}$ if Π is also *honest-verifier zero-knowledge (HVZK)* against sub-exponential size distinguishers and has "short" challenges. The above includes many natural Σ-protocols.

As an additional application of our result on two-round zero knowledge, we show a black-box impossibility result for proving soundness of Micali's CS-proofs [Mic94] based on any falsifiable assumption. We note that unlike [GW11], this result also holds for *non-adaptive* cheating proves, who choose the instance before seeing the verifier's message.

We wish to emphasize that these results do *not* refute the highly believable conjecture that the FS heuristic can be securely instantiated for all proofs and many natural arguments. However, it shows that we will need to rely on new "non-standard" assumptions or develop new "non-black box" proof techniques if we ever hope to prove this conjecture.

Assumptions. To capture all "standard assumptions", we consider general classes of assumptions defined in terms of the syntactic format that the assumption takes. A "*cryptographic game assumption*" has the format of an interactive game between a (possibly inefficient) challenger who interacts in a black-box manner with some candidate attacker. The assumption states that every efficient attacker has at most negligible probability in winning this game. This notion is due to [DOP05, HH09]. A "*falsifiable assumption*" [Nao03] is a cryptographic game assumption where the challenger is also *efficient*. Note that these notions capture essentially all of the concrete assumptions we use in cryptography, such as the hardness of factoring, the RSA problem, the discrete logarithm problem, the computational/decissional Diffie-Hellman problem (CDH/DDH), learning with errors (LWE), etc. We stress that these notions are defined as liberally as possible so as to include essentially everything that could be considered a "standard assumption", and to make our negative result as strong as possible. Of course, it may also capture many non-standard (and false) assumptions, as well as trivially true and uninteresting assumptions.

FS-Universality. The assumption that a hash function is FS-universal does not have the format of a *cryptographic game*, since the assumption quantifies over all proof systems. In particular, an attack against "FS-universality" consists of two components: a 3PC proof system $\Pi = (P, V)$ for some language $\mathcal{L}$ and a breaker $\mathcal{A}$ that breaks the soundness of the Fiat-Shamir transform applied to Π. The challenger cannot test that Π is a 3PC proof system by interacting with P, V in a black-box manner. When we talk about *black-box reductions* for FS-universality, we naturally restrict the challenger to interact with $P, V, \mathcal{A}$ as a black box. In other words, the reduction is black-box in the code of the attacker, as well as the proof system Π.

FS(Π)-Security. For a particular proof system Π for a language $\mathcal{L}$, the assumption that a hash function is FS(Π)-secure *is* a cryptographic game assumption: the attacker wins if he can come up with a false statement x and an accepting proof π under the non-interactive argument that we get by applying the FS heuristic to Π. However, it does not have the format of a *falsifiable assumption* since the challenger cannot efficiently test whether x is false statement, and therefore, whether the attacker breaks soundness.

2 Preliminaries and Definitions

Let n denote the security parameter. We say that a function $f(n) = 1/n^{\omega(1)}$ *negligible* in the security parameter, and denote it by $\mathsf{negl}(n)$. We consider the class of efficient schemes to be ones that can be implemented by a probabilistic

polynomial-time Turing machine, denoted by PPT. In contrast, we consider the class of efficient adversaries $\mathcal{A} = \{\mathcal{A}_n\}$ to be non-uniform families of polynomial-size circuits, denoted by $\mathsf{polysize}$.

We start by describing the Fiat-Shamir heuristic for public-coin interactive proofs. Recall that an interactive proof system [GMR89] for a language L with corresponding relation R is a tuple of efficient algorithms $\Pi = (\mathcal{P}, \mathcal{V})$, where $\mathcal{P}$ and $\mathcal{V}$ denote the prover and the verifier algorithms respectively. We assume familiarity of the reader with the standard notions of *completeness* and *soundness* for an interactive proof system, and skip formal defnitions.

The Fiat-Shamir Heuristic. Throughout the paper, we will mainly focus on the special case of applying the FS heuristic to a *3-message public-coin (3PC)* interactive proof system $\Pi = \langle P, V\rangle$ for an NP relation $\mathcal{R}$.[1] Denote the first message of the prover by α, the verifier's challenge by β, and the final message of the prover by γ. Also, let $\pi = (\alpha, \beta, \gamma)$ denote the transcript of the execution.

For security parameter n, let $m(n)$ and $k(n)$ denote the lengths of α and β, respectively. Let $\mathcal{H} = \{h_s : \{0,1\}^{m(n)} \to \{0,1\}^{k(n)}\}_{n\in\mathbb{N}, s\in\{0,1\}^{\ell(n)}}$ be a family of hash functions mapping m bits to k bits. The $\mathsf{Fiat\text{-}Shamir\ collapse}$ (or FS-collapse in short) of protocol $\Pi = \langle P, V\rangle$ using $\mathcal{H}$ is a two-message protocol $\Pi^{\mathsf{FS}} = \langle P_{\mathsf{FS}}, V_{\mathsf{FS}}\rangle$ defined as follows:

- In the first message, the FS verifier $V_{\mathsf{FS}}(1^n, x)$ selects a random seed $s \leftarrow \{0,1\}^{\ell(n)}$ for the hash function. (We can also skip this step by thinking of s as a common reference string).
- In the second message, the FS prover $P_{\mathsf{FS}}(1^n, x, w)$ runs $P(1^n, x, w)$ to derive its first message α. It then computes the challenge $\beta := h_s(\alpha)$ by hashing α, and passes β to P to get its third message γ. Finally, P_{FS} outputs the tuple (α, β, γ).
- The FS verifier $V_{\mathsf{FS}}(1^n, x)$ accepts the proof if $\beta = h_s(\alpha)$ *and* the original verifier $V(1^n, x)$ accepts the protocol (α, β, γ) when executed with random-coins β.

We say that the *FS-collapse is sound* if the resulting protocol Π^{FS} is a *computationally-sound argument system* as specified below.

Definition 1 (Fiat-Shamir soundness). *We say that Π^{FS} is computationally sound if, for any* $\mathsf{polysize}$ *prover $P^* = \{P^*_n\}$ and $x \notin \mathcal{L}(\mathcal{R})$*

$$\Pr_{s\xleftarrow{\$}\{0,1\}^{\ell(n)}}\left[V(1^n, x, \pi) = 1 \;\middle|\; \begin{array}{c}\pi \leftarrow P^*_n(x, s)\\ \pi = (\alpha, \beta, \gamma)\\ h_s(\alpha) = \beta\end{array}\right] \leq \mathsf{negl}(n) \ .$$

We call the above probability the advantage *of P^* in breaking computational soundness.*

[1] Indeed, this is the most common but also minimal case for which Fiat-Shamir is expected to work, and therefore restricting ourselves to this case gives us the strongest negative result.

Cryptographic Games and Falsifiable Assumptions. Cryptographic games present a general framework for defining cryptographic assumptions and security properties. A game is given by a protocol specified via a *challenger* who interacts with an arbitrary *attacker* – security mandates that no efficient attacker should be able to win the game with better than negligible probability.

Definition 2 (Cryptographic game [HH09]). *A* cryptographic game $\mathcal{G} = (\Gamma, c)$ *is defined by a (possibly inefficient) random system* Γ*, called the* challenger*, and a constant* $c \in [0, 1)$*. On security parameter* n*, the challenger* $\Gamma(1^n)$ *interacts with some attacker* $\mathcal{A}_n$ *and outputs a bit* b*. We denote the output of this interaction by* $b = (\mathcal{A}_n \leftrightarrows \Gamma(1^n))$*. The* advantage *of an attacker* $\mathcal{A}_n$ *in the game* $\mathcal{G}$ *is defined as*

$$\mathbf{Adv}^{\mathcal{A}}_{\mathcal{G}}(n) \stackrel{def}{=} \Pr[\ (\mathcal{A}_n \leftrightarrows \Gamma(1^n)) = 1\ ] - c\ .$$

A cryptographic game $\mathcal{G}$ *is* secure *if for all* polysize *attackers* $\mathcal{A} = \{\mathcal{A}_n\}$*, the advantage* $\mathbf{Adv}^{\mathcal{A}}_{\mathcal{G}}(n)$ *is negligible. The game is* $T(n)$*-secure if for all attackers running in time* $\mathsf{poly}(T(n))$ *the advantage* $\mathbf{Adv}^{\mathcal{A}}_{\mathcal{G}}(n)$ *is* $\mathsf{negl}(T(n)) = T(n)^{-\omega(1)}$*.*

When $c = 0$, the above definition of cryptographic games captures *search problems* such as factoring, the discrete logarithm problem, signature security etc. When $c = \frac{1}{2}$, it captures *decisional problems* such as DDH, encryption security etc. Note that cryptographic games may be highly interactive and may not even have any a-priori bound on the number of rounds of interaction between $\mathcal{A}$ and Γ. The work of [GW11] defined a more restricted notion of cryptographic games called "falsifiable assumptions" (following [Nao03]) where the challenger is also required to be efficient.

Definition 3 (Falsifiable Assumption). *We say that a cryptographic game* $\mathcal{G} = (\Gamma, c)$ *is a falsifiable assumption if the challenger* $\Gamma(1^n)$ *runs in time* $\mathsf{poly}(n)$*.*

3 Black-Box Impossibility of Entropy-Preserving Hashing and Fiat-Shamir Universality

In this section, we show a black-box separation between hash function that are *Fiat-Shamir-universal* and general cryptographic games. As explained in the introduction, an FS-universal hash function family guarantees the soundness of the Fiat-Shamir heuristic for *any* 3PC system with appropriate message and challenge length.

Definition 4 ((m, k)-FS-universal hash function). *We say that a hash-function family* $\mathcal{H} = \{h_s : \{0,1\}^{m(n)} \to \{0,1\}^{k(n)}\}_{s \in \{0,1\}^{\ell(n)}}$ *is* $(m(n), k(n))$*-FS-universal if for every 3PC (statistically sound) proof system* $\langle P, V \rangle$ *with first and second messages of respective lengths* $m = m(n)$ *and* $k = k(n)$*, the FS-collapse* Π^{FS} *is a (computationally sound) argument.*

As the main step towards this separation, we show a black-box separation between the notion of entropy-preserving hash-functions introduced by Barak et al. [BLV03] and general cryptographic games. We then leverage the connection between entropy-preserving hashing and FS-universal hashing as shown in [BLV03, DRV12] to prove a similar seperation for the latter.

3.1 Black-Box Impossibility for Entropy-Preserving Hashing

Barak et al. [BLV03] formulated a relatively simple entropy preservation property for hash functions, and showed that it is sufficient for FS-universality. Recall that the *(Shannon) entropy* of a random variable $\mathbf{x}$ is $\mathbf{H}(\mathbf{x}) = \mathbf{E}_{x \xleftarrow{\$} \mathbf{x}}[-\log(\Pr[\mathbf{x} = x])]$. For jointly distributed random variables $(\mathbf{x}, \mathbf{y})$, the *conditional entropy* of $\mathbf{x}$ given $\mathbf{y}$ is defined by

$$\mathbf{H}(\mathbf{x} \mid \mathbf{y}) = \mathop{\mathbf{E}}_{y \xleftarrow{\$} \mathbf{y}} [\mathbf{H}(\mathbf{x} \mid \mathbf{y} = y)],$$

where $\mathbf{x}|_{\mathbf{y}=y}$ is a random variable distributed according to $\mathbf{x}$ conditioned on $\mathbf{y} = y$.

Definition 5 (Definition 9.2 in [BLV03]). *We say that a hash function family* $\mathcal{H} = \{h_s : \{0,1\}^{m(n)} \to \{0,1\}^{k(n)}\}_{s \in \{0,1\}^{\ell(n)}}$ preserves $u(n)$-entropy*, if for any* polysize $\mathcal{A}$*, and all large enough values of the security parameter* $n \in \mathbb{N}$ *we have*

$$\mathbf{H}(h_{\mathbf{s}}(\mathbf{x}) \mid \mathbf{x}) > u(n) \ ,$$

where $\mathbf{s}, \mathbf{x}$ *are correlated random variables defined by choosing* $\mathbf{s}$ *uniformly at random over* $\{0,1\}^{\ell(n)}$*, and setting* $\mathbf{x}$ *to be the first* $m(n)$ *bits of the output of* $\mathcal{A}(1^n, \mathbf{s})$*. We say that the hash function (just plain)* preserves entropy *if it preserves* $u(n)$*-entropy for* $u(n) = 0$*.*

The work of [BLV03] shows that any hash function family that preserves $u(n) = k(n) - O(\log n)$ entropy is (m, k)-FS-universal. An alternative take on the notion of "entropy preserving" hash functions and a detailed exploration of the parameters is given by Dodis, Ristenpart, and Vadhan [DRV12]. The same work also shows an implication in the reverse direction: any (m, k)-FS-universal hash function family must also preserve entropy. We will thus focus on showing a black-box separation for entropy-preserving hash functions, and then adapt the [DRV12] result to our setting.

Black-Box Reductions. We now define the notion of a black-box reduction from entropy-preserving hashing to a cryptographic game.

Definition 6 (BB Reduction for Entropy Preserving Hash). *Let* $\mathcal{G} = (\Gamma, c)$ *be a cryptographic game and let* $\mathcal{H}$ *be a hash function family with input length* $m(n)$ *and output length* $k(n)$*, for some polynomials* m, k*. A* black-box reduction showing that $\mathcal{H}$ is entropy-preserving from the security of the game $\mathcal{G}$ *is an oracle-access* PPT *machine* $\mathcal{B}^{(\cdot)}$ *for which there exists some polynomial* p *such that the following holds. Let* $\mathcal{A} = \{\mathcal{A}_n\}$ *be any (possibly inefficient) attacker*

such that $\mathbf{H}(h_{\mathbf{s}}(\mathbf{x}) \mid \mathbf{x}) = 0$*, where the random variable* $\mathbf{s}, \mathbf{x}$ *are defined the same way as in Definition 5, i.e.,* $\mathbf{s} \xleftarrow{\$} \{0,1\}^{\ell(n)}$*, and* $\mathbf{x} \leftarrow \mathcal{A}_n(\mathbf{s})$*. Then, the advantage of* $\mathcal{B}^{\mathcal{A}_n}(1^n)$ *in the game* $\mathcal{G}$ *is at least* $1/p(n)$*.*

Remark 1 (Reductions from $T(n)$-security assumptions). We can also consider a variant, where the black-box reduction is from the $T(n)$-security of the cryptographic game $\mathcal{G}$. In this case, we allow the reduction $\mathcal{B}^{(\cdot)}$ to run in time $\mathsf{poly}(T(n))$ and only insist that its advantage is $\geq 1/p(T(n))$.

For simplicity, we insist that the reduction itself has some *noticeable* advantage $1/p(n)$ rather than the standard requirement that its advantage is simply *non-negligible*. Furthermore, we also insist that the reduction is *security-parameter preserving* meaning that when it is called with security parameter 1^n it only accesses the oracle $\mathcal{A}_n$ on the *same* security parameter n. The above two requirements come with some loss of generality, but they hold for all of the natural reductions in cryptography.

BB Separation via Simulatable Attack. We now outline a general strategy for proving black-box separations via a technique called a *simulatable attack*. This strategy has been used in several prior works [BV98, Cor02, Bro05, PV05, GBL08] [DOP05, HH09, GW11, Pas11, Seu12, DHT12, Wic12]. The main idea of this paradigm is to construct a special *inefficient* attacker $\mathcal{A}$ that breaks the security of the target primitive (in our case, the entropy-preserving security of $\mathcal{H}$), but for which there is an *efficient* simulator Sim such that no distinguisher can tell the difference between "black-box" interaction with Sim and $\mathcal{A}$. This means that any efficient black-box reduction which can win some cryptographic game, given oracle access to the inefficient attacker $\mathcal{A}$, can also win the cryptographic game, given oracle access to the efficient simulator Sim. Hence, if we have a black-box reduction showing the entropy-preserving security of $\mathcal{H}$ under some cryptographic-game assumption, it implies that the reduction, together with the efficient simulator Sim, give us an efficient stand-alone attack against the assumption, and so it cannot be secure to begin with!

Aspects of this technique were recently formalized in [Wic12], and we will rely on the notation and the results from that work. However, for concreteness, we only restrict ourselves to describing this strategy for the specific case of *entropy preserving hash functions*.

Definition 7 (Simulatable Attack for Entropy-Preserving Hashing). *Let $\mathcal{H}$ be some hash function family with input length $m(n)$ and output length $k(n)$. A* $\varepsilon(n)$-simulatable attack *on the entropy-preserving security of $\mathcal{H}$ consists of: (1) an ensemble of (possibly inefficient) stateless non-uniform attackers $\{\mathcal{A}_{n,f}\}_{n\in\mathbb{N}, f\in\mathcal{F}_n}$ where $\{\mathcal{F}_n\}$ is some ensemble of finite sets, and (2) a stateful* PPT *simulator* Sim*. We require that the following two properties hold:*

- *For each $n \in \mathbb{N}, f \in \mathcal{F}_n$, the (inefficient) attacker $\mathcal{A}_{n,f}$ successfully breaks the entropy-preserving security of $\mathcal{H}$.*

– *For every (possibly inefficient) oracle access machine $\mathcal{M}^{(\cdot)}$, making at most $q = q(n)$ queries to its oracle:*

$$\left| \Pr_{f \xleftarrow{\$} \mathcal{F}_n, \mathcal{M}} [\mathcal{M}^{\mathcal{A}_{n,r}}(1^n) = 1] \; - \; \Pr_{(\mathcal{M}, \mathsf{Sim})} [\mathcal{M}^{\mathsf{Sim}(1^n)}(1^n) = 1] \right| \leq \mathsf{poly}(q(n)) \cdot \varepsilon(n).$$

namely, oracle access to $\mathcal{A}_{n,f}$ for a random $f \xleftarrow{\$} \mathcal{F}_n$ is indistinguishable from that to Sim.

We omit the $\varepsilon(n)$ and just say "simulatable attack" as shorthand for an $\varepsilon(n)$-simulatable attack with some negligible $\varepsilon(n) = \mathsf{negl}(n)$.

As discussed in the introduction, the existence of a simulatable attack against some scheme $\mathcal{H}$ ensures that one cannot prove the security of $\mathcal{H}$ using black-box reduction from cryptographic game assumption, unless the assumption is false. This is because a reduction must be able to use the simulatable attacker $\mathcal{A}$ against $\mathcal{H}$ to break the underlying assumption, but then this means that the reduction and the simulator together would give us an efficient stand-alone attack against the assumption to begin with. A general version of this theorem was given in [Wic12] and therefore we get the following as a special case.

Theorem 1 (Special case of [Wic12]). *If there exists a simulatable attack against the entropy preserving security of $\mathcal{H}$, and there is a black-box reduction showing the entropy preserving security of $\mathcal{H}$ from the security of some cryptographic game $\mathcal{G}$, then $\mathcal{G}$ is* not *secure.*

Furthermore, for any $T(n)$, if there exists an $\varepsilon(n) = T(n)^{-\omega(1)}$-simulatable attack against $\mathcal{H}$ and there is a black-box reduction from the $T(n)$-security of $\mathcal{G}$, then $\mathcal{G}$ is not *$T(n)$-secure.*

Constructing a Simulatable Attack. We now show that, for any family of hash functions $\mathcal{H}$, there is a simulatable attack against its entropy preserving security.

Theorem 2. *Let $\mathcal{H} = \{h_s : \{0,1\}^{m(n)} \to \{0,1\}^{k(n)}\}_{n \in \mathbb{N}, s \in \{0,1\}^{\ell(n)}}$ be any family of hash functions. Then there is a $2^{-\Omega(m-k)}$-simulatable attack against the entropy preserving security of $\mathcal{H}$.*

Proof outline. Let $\mathcal{F}_n$ be the set of functions $f : \{0,1\}^{m(n)} \to \{0,1\}^{k(n)}$, and let $\mathcal{F}_n^* \subseteq \mathcal{F}_n$ be a subset consisting of all the functions f such that for every $s \in \{0,1\}^{\ell(n)}$, there is some $x \in \{0,1\}^m$ on which $h_s(x) = f(x)$. We will define a family of inefficient attackers $\{\mathsf{Break}_f\}$, indexed by functions $f \in \mathcal{F}_n^*$, that break the entropy preserving security of $\mathcal{H}$. Before we do so, we first note that a simple counting argument shows that $\mathcal{F}_n^*$ is non-empty, and in fact forms a very dense subset of $\mathcal{F}_n$.

Claim. $\mathcal{F}_n^*$ is dense in $\mathcal{F}_n$ with $\frac{|\mathcal{F}_n^*|}{|\mathcal{F}_n|} = (1 - 2^{-\Omega(2^{m-k})})$-fraction of $\mathcal{F}_n$.

$\mathsf{Break}_f : f \in \mathcal{F}_n^*$

Given input $s \in \{0,1\}^{\ell(n)}$, output a random x from the set of all values satisfying $h_s(x) = f(x)$.
(By the definition of $\mathcal{F}_n^*$, at least one such x always exists.)

Fig. 1.

Constructing an attack. Now, we are ready to define a family of inefficient attackers $\{\mathsf{Break}_f\}$, indexed by functions $f \in \mathcal{F}_n^*$, that break the entropy preserving security of $\mathcal{H}$ as follows:

The attack is successful. For any fixed $f \in \mathcal{F}_n^*$, it is easy to see that the attacker Break_f breaks the entropy preserving security of $\mathcal{H}$. This is because, conditioned on seeing any output $x \leftarrow \mathsf{Break}_f(s)$, we can completely determine the value $h_s(x)$ without knowing the seed s, via the relation $h_s(x) = f(x)$. Therefore, defining the random variables $\mathbf{s}$ to be uniform over $\{0,1\}^{\ell(n)}$ and $\mathbf{x} \leftarrow \mathsf{Break}_f(\mathbf{s})$, we have $\mathbf{H}(h_{\mathbf{s}}(\mathbf{x}) \mid \mathbf{x}) = 0$ as desired.

The simulator for the attack. The more interesting part of the proof is showing that for random $f \leftarrow \mathcal{F}_n^*$, the attacker Break_f can be simulated very efficiently, with a small statistical error. Our (stateful) simulator is incredibly simple and, on each invocation, just outputs a fresh random value (which wasn't output previously). It is easy to see that the simulator satisfies the efficiency requirements

$\mathsf{Sim}(1^n)$

Initialize the set $X := \emptyset$.
On input $s \in \{0,1\}^{\ell(n)}$: Sample $x \leftarrow \{0,1\}^m \setminus X$, add x to the set X, and output x.

Fig. 2.

of the definition of a simulatable attack.

Indistinguishability of simulator. The next step is to show that a random attacker from the class $\{\mathsf{Break}_f\}$ and the above simulator are statistically indistinguishable. In particular, for any (computationally unbounded) q-query distinguisher $\mathcal{M}$,

$$\left| \Pr_{f \xleftarrow{\$} \mathcal{F}_n^*} \left[\mathcal{M}^{\mathsf{Break}_f}(1^n) = 1\right] - \Pr_{\mathsf{Sim}} \left[\mathcal{M}^{\mathsf{Sim}(1^n)}(1^n) = 1\right] \right| \leq q^2 \cdot 2^{-\Omega(m-k)} \ .$$

Theorem 1 and Theorem 2 allow us to conclude the following.

Corollary 1. *Let $\mathcal{G} = (\Gamma, c)$ be a cryptographic game assumption and let $\mathcal{H}$ be an (m,k)-hash function family for some polynomials $m = m(n), k = k(n)$ such that $m(n) - k(n) = \omega(\log(n))$. If there is a black-box reduction showing that*

$\mathcal{H}$ is entropy-preserving from the security of the game $\mathcal{G}$, then $\mathcal{G}$ is not secure. Furthermore, if $m(n) - k(n) = \omega(\log(T(n))$ and there is a black-box reduction showing that $\mathcal{H}$ is entropy preserving from the $T(n)$-security of $\mathcal{G}$, then $\mathcal{G}$ is not $T(n)$-secure.

3.2 Black-Box Impossibility of Fiat-Shamir Universality

As we have already mentioned, the work of Dodis, Ristenpart and Vadhan [DRV12], shows that any FS-universal hash function family $\mathcal{H}$ must also be entropy-preserving. Intuitively, this should imply that our negative result for entropy-preserving hashing from the previous section should yield a similar negative result for FS-universal hashing. Indeed, we do show a theorem along these lines. However, formalizing the above intuition requires some care. For example, it becomes important that our notion of black-box reductions for FS-universal hashing treats the 3PC proof-system as a black box. Intuitively, this is because the result of [DRV12] uses the attacker $\mathcal{A}$ against the entropy-preserving security of a hash family $\mathcal{H}$ to *construct* a 3PC proof system $\Pi^{\mathcal{A}} = \langle P^{\mathcal{A}}, V^{\mathcal{A}} \rangle$ as well as to *attacker* $\mathcal{D}^{\mathcal{A}}$ that breaks the soundness of the FS-collapse of $\Pi^{\mathcal{A}}$. Therefore, any black-box reduction that shows the FS-universality of $\mathcal{H}$ under some game assumption by treating the proof system $\Pi^{\mathcal{A}} = \langle P^{\mathcal{A}}, V^{\mathcal{A}} \rangle$ and the attacker $\mathcal{D}^{\mathcal{A}}$ as a black box, can also be used as a reduction showing the entropy-preserving security of $\mathcal{H}$ under the same assumption by treating the attacker $\mathcal{A}$ as a black box. Further details can be found in [BGW12].

4 Impossibility of Fiat-Shamir for Specific Proof Systems

In this section, we show that for many well-studied public-coin interactive proofs, the soundness of the Fiat-Shamir heuristic cannot be proven via a black-box reduction to any falsifiable assumption. Using similar techniques, we also show a black-box impossibility result for proving soundness of Micali's CS-proofs [Mic94] based on any falsifiable assumption. The main tool underlying both of these results is a black-box impossibility result for two-round zero-knowledge w.r.t. super-polynomial simulation.

We note that the connection between zero-knowledge and the (in)security of Fiat-Shamir heuristic was already made in prior works. In particular, Dwork et al. [DNRS99] showed that if a public-coin interactive protocol is "weakly" zero-knowledge (where the ZK property is weakened by changing the order of quantifiers in the standard ZK definition, but requiring the simulator and distinguisher to be polynomial time) then the Fiat-Shamir heuristic applied to this protocol is not sound. We note however, that known public-coin protocols where the FS-heuristic would typically be applied, are *not* known to satisfy their zero-knowledge property. In contrast, (as we discuss below) we only require the protocol to be honest-verifier zero-knowledge w.r.t. sub-exponential adversaries, and show that this property is satisfied by many well-known protocols (under some assumptions).

The rest of this section is organized as follows. In Section 4.1, we prove a general theorem on the black-box impossibility of 2-round zero-knowledge arguements. In Section 4.2 we apply this theorem to show that for many well-studied public-coin interactive *proofs*, the soundness of the Fiat-Shamir heuristic cannot be proven via a black-box reduction to any falsifiable assumption. Finally, in Section 4.3, we extend our techniques to show a black-box impossibility result for proving soundness of Micali's CS-proofs [Mic94].

4.1 Black-Box Impossibility for 2-Round Zero Knowledge

In this section, we give a black-box impossibility result for 2-round zero-knowledge arguments. Our theorem extends the negative result of Goldreich and Oren [GO94], and can be seen as essentially tight, in view of the positive result of Pass [Pas03]. We refer the reader to the full version [DJKL12] for a detailed comparison of our result with [GO94] and [Pas03].

We start with some preliminaries and then describe our result.

Hard Languages and Zero-Knowledge Proofs. We start by formally defining a hard NP language.

Definition 8 (T-Hard Language). *For any $T = T(n)$, an* NP *language L is said to be* T-hard *if there exist two distribution families $\mathcal{X} = \{\mathcal{X}_n\}_{n\in\mathbb{N}}$ and $\bar{\mathcal{X}} = \{\bar{\mathcal{X}}_n\}_{n\in\mathbb{N}}$, and a PPT sampling algorithm* Samp *such that:*

- *For every $n \in \mathbb{N}$ the support of $\mathcal{X}_n$ is in L and the support of $\bar{\mathcal{X}}_n$ is in $\bar{L}$.*
- *The distributions $\mathcal{X}$ and $\bar{\mathcal{X}}$ are $T(n)$-indistinguishable.*
- *The support of the sampling algorithm* Samp *consists of elements (x, w) such that $R(x, w) = 1$, and its projection to the first coordinate yields the distribution $\mathcal{X} = \{\mathcal{X}_n\}_{n\in\mathbb{N}}$.*
  Note that since Samp *is efficient, the distribution family $\mathcal{X}$ is efficiently sampleable. There are no constraints on the size of the instances in $\mathcal{X}_n$ or $\bar{\mathcal{X}}_n$, however since $\mathcal{X}$ is efficiently sampleable each $x \leftarrow \mathcal{X}_n$ is of size at most* $\mathrm{poly}(n)$.

An NP *language is said to be* sub-exponentially hard *if it is 2^n-hard.*[2]

We now define the zero-knowedge property for an interactive proof system [GMR89].

Definition 9 (T-Zero Knowledge). *For any $T = T(n)$, we say that an interactive proof system $\Pi = (\mathcal{P}, \mathcal{V})$ for an* NP *language L is* (auxiliary-input) T-zero-knowledge *if for every poly-size circuit $\mathcal{V}^*$ there exists a simulator $\mathcal{S}_{\mathcal{V}^*}(1^n)$ of size* $\mathrm{poly}(T(n))$ *such that for every $n \in \mathbb{N}$, every instance $x \in L$ of length*

[2] Note that it should be hard for a $\mathrm{poly}(2^n)$-time distinguisher to distinguish between elements in $\mathcal{X}_n$ and elements in $\bar{\mathcal{X}}_n$, where these elements can be much longer than n, and can be of length n^ϵ for any constant $\epsilon > 0$ (thus, capturing the *sub*-exponential hardness).

at most $\text{poly}(n)$ *with a corresponding witness* w*, and every auxiliary input* $z \in \{0,1\}^{\text{poly}(n)}$*, it holds that for every non-uniform distinguisher* $D = \{D_n\}$ *of size* $\text{poly}(T(n))$

$$\left|\Pr[D\left((\mathcal{P}(w), \mathcal{V}^*(z))(1^n, x)\right) = 1] - \Pr[D\left(\mathcal{S}_{\mathcal{V}^*}(1^n, x, z)\right) = 1]\right| \leq \text{negl}(T(n)),$$

where $(\mathcal{P}(w), \mathcal{V}^*(z))(1^n, x)$ *denotes the view of the verifier* $\mathcal{V}^*$ *after interacting with the honest prover on input security parameter* n*, statement* $x \in L$*, auxiliary input* z*, and* $\mathcal{S}_{\mathcal{V}^*}(1^n, x, z)$ *denotes the output of the simulator* $\mathcal{S}_{\mathcal{V}^*}$ *on input* $(1^n, x, z)$*.*

We now state our main technical theorem:

Theorem 3. *For any* $T(n)$ *and any* T*-hard language* L*, there does not exist a 2-round argument system* Π *for* L *such that:*

- Π *is (auxiliary-input)* T*-zero-knowledge, and*
- *the soundness of* Π *can be proven via a black-box reduction to a* T*-hard falsifiable assumption,*

unless the assumption is false.

Theorem 3, which we believe to be of independent interest, is also the starting point for our impossibility results for the Fiat-Shamir paradigm (see Section 4.2) and for CS proofs (see Section 4.3).

Proof Idea. Consider a 2-round argument system Π for a T-hard language L that is (auxiliary-input) T-zero-knowledge. We prove, by contradiction, that the soundness of Π cannot be proven via a black-box reduction to a T-hard falsifiable assumption. Let n be a security parameter and suppose that there exists a $\text{poly}(T(n))$-time black-box reduction $\mathcal{R}$ such that given black-box oracle access to any cheating prover $\mathcal{P}^*$, uses this oracle to break a $T(n)$-hard falsifiable assumption. By the definitions of a $T(n)$-hard falsifiable assumption and a black-box reduction, we know the reduction $\mathcal{R}$ runs in time $\text{poly}(T(n))$.

By naturally extending Goldreich and Oren's 2-round zero-knowledge impossibility result [GO94], we first prove that the T-zero-knowledge simulator $\mathcal{S}$ always produces an accepting transcript, even when given a statement $x \in \bar{L}$. Thus, we may view $\mathcal{S}$ as a cheating prover. This means that $\mathcal{R}$ breaks the assumption when given oracle access to $\mathcal{S}$ (and $\mathcal{S}$ is given $x \in \bar{L}$). For brevity, we say that $\mathcal{R}^{\mathcal{S}(x \in \bar{L})}$ breaks the assumption. However, we must be careful because the reduction $\mathcal{R}$ may "lie" about the security parameter and run $\mathcal{S}$ with security parameter $\kappa \neq n$. We denote by n the security parameter of the underlying falsifiable assumption, and denote by κ the security parameter that the reduction uses when calling $\mathcal{S}$ (though the reduction $\mathcal{R}$ may call $\mathcal{S}$ many times with different security parameters). Note that the bound on the running time of $\mathcal{R}$ means $\kappa \leq T(n)$.

Our approach is to show that oracle access to $\mathcal{S}(x \in \bar{L}_\kappa)$ can be simulated in time $\text{poly}(T(n))$ regardless of the value of κ. If $\kappa \leq n$ then $\mathcal{S}(x \in \bar{L}_\kappa)$ runs in time $\text{poly}(T(\kappa)) \leq \text{poly}(T(n))$ and we are done. However, if $\kappa > n$ then we show

that if $\mathcal{R}^{\mathcal{S}(x\in\bar{L}_\kappa)}$ breaks the assumption then so does $\mathcal{R}^{\mathcal{P}(x\in L_\kappa,w)}$, where w is a valid witness for $x \in L_\kappa$ and $\mathcal{P}$ is the honest prover. Since $\mathcal{P}(x \in L_\kappa, w)$ runs in time $\text{poly}(\kappa) \leq \text{poly}(T(n))$, this means we can simulate $\mathcal{S}(x \in \bar{L}_\kappa)$ in time $\text{poly}(T(n))$.

4.2 Black-Box Impossibility for Fiat-Shamir Paradigm

For the sake of simplicity of notation, we present our results for the case of 3-round public-coin protocols. We note that although our techniques generalize to constant-round protocols, the case of 3-rounds already covers many interesting applications of the Fiat-Shamir paradigm.

We start by defining *special honest-verifier (auxiliary-input) T-zero-knowledge*. We will later show the black-box impossibility results for protocols which have this property.

Definition 10. *For any $T = T(n)$, we say that a 3-round public-coin proof (or argument) system $\Pi = (\mathcal{P}, \mathcal{V})$ for an* NP *language L is* (auxiliary-input) special honest-verifier T-zero-knowledge *if there exists a simulator $\mathcal{S}(1^n)$ of size* $\text{poly}(T(n))$ *such that for every $n \in \mathbb{N}$, every instance $x \in L$ of length at most* $\text{poly}(n)$ *with a corresponding witness w, every auxiliary input $z \in \{0,1\}^{\text{poly}(n)}$, and every random tape β of the verifier it holds that for every non-uniform distinguisher $D = \{D_n\}$ of size* $\text{poly}(T(n))$

$$\left|\Pr[D\left((\mathcal{P}(w), \mathcal{V}(z,\beta))(1^n, x)\right) = 1] - \Pr[D\left(\mathcal{S}(1^n, x, z, \beta)\right) = 1]\right| \leq \text{negl}(T(n)),$$

where $(\mathcal{P}(w), \mathcal{V}(z,\beta))(1^n, x)$ denotes the view of the honest verifier $\mathcal{V}$ after interacting with the honest prover on input security parameter n, statement $x \in L$, auxiliary input z, and random tape β, and $\mathcal{S}(1^n, x, z, \beta)$ denotes the output of the simulator $\mathcal{S}$ on the corresponding inputs.

We note that special honest verifier zero knowledge differs from honest verifier zero knowledge since the simulator must successfully simulate the view of the honest verifier for *every* given random tape β.

We now state the main theorem of this section:

Theorem 4. *For any $T(n)$ and any T-hard language L, let Π be a 3-round public-coin proof (or argument) system for $\mathcal{L}$ with $2^{|\beta|} \leq T(n)$ which is special honest verifier (auxiliary input) T-zero knowledge. Then, the soundness of the FS-collapse of Π, namely, Π^{FS}, cannot be proven via a black-box reduction to a T-hard falsifiable assumption (unless the assumption is false).*

Note that many public-coin proof (or argument) systems (such as those discussed in Section 4.2) consist of ℓ parallel repetitions of a basic protocol where the length of the verifier's message is a constant number of bits (or may depend logarithmically on the size of the instance x). To save on communication, it is desirable to repeat the protocol only $\ell = \text{poly}\log(n)$ times, since this already achieves negligible soundness error. For such protocols, Theorem 4 implies that if

the language L is quasi-polyomially hard, then the Fiat-Shamir transformation applied to this protocol cannot be proven sound via a black-box reduction to a falsifiable assumption.

Given Theorem 4, one may hypothesize that the Fiat-Shamir transformation, when applied to protocols of the type discussed above, can in fact be proven secure (via a black-box reduction to a falsifiable assumption) when the number of parallel repetitions is increased to $\ell = \text{poly}(n)$. However, we show that this is not the case; for many protocols of interest, the impossibility result holds even when the number of repetitions ℓ, is greater than the hardness of the language.

Corollary 2. *Let L be a sub-exponentially hard language and let Π be a 3-round public-coin proof (or argument) system for L with the following properties:*

- *The length of the second message, β, is polynomial in the security parameter, n, and is independent of the length of the instance, x.*
- *Π is special honest verifier (auxiliary input) $2^{|\beta|}$-zero knowledge.*

Then, the soundness of the FS-collapse of Π, namely, Π^{FS}, cannot be proven via a black-box reduction to a $2^{|\beta|}$-hard falsifiable assumption (unless the assumption is false).

Corollary 2 follows from Theorem 4, as follows. Recall that a language is said to be sub-exponentially hard if it is T-hard for $T(n) = 2^n$ (see Definition 8). Namely, if there exist distributions $\mathcal{X}_n$ and $\bar{\mathcal{X}}_n$ over strings of length $\text{poly}(n)$ that are 2^n-indistinguishable, where $\mathcal{X}_n$ is a distribution over instances in the language and $\bar{\mathcal{X}}_n$ is a distribution over instances outside the language. Note that the length of these instances can be much larger than n, and can be of length $n^{1/\epsilon}$ for any constant $\epsilon > 0$.

We argue that any sub-exponentially hard language is also $2^{p(n)}$-hard, for any polynomial p. This follows by simply taking $\mathcal{X}'_n = \mathcal{X}_{p(n)}$ and by taking $\bar{\mathcal{X}}'_n = \bar{\mathcal{X}}_{p(n)}$. Using this observation, Corollary 2 follows immediately from Theorem 4 by choosing $T(n) = 2^{p(n)}$ such that $|\beta| = p(n)$.

Remark 2. It was first observed by Dwork et al. [DNRS99] that if Π is a 3-round public-coin proof (or argument) system for $\mathcal{L}$ that is T-zero-knowledge for $T = \text{poly}(n)$, then the transformed protocol, Π^{FS}, cannot be not sound. In contrast, we prove our results for protocols Π that have *inefficient* zero-knowledge simulators; i.e., simulators that run in T-time, where T is superpolynomial in n. Note, however, that we only require *standard* soundness from Π^{FS}; i.e., we require that Π^{FS} is sound against *efficient*, polynomial-time, adversaries. Thus, our results do not follow from [DNRS99].

Applications of Theorem 4 and Corollary 2. Typically (or at least traditionally), the Fiat-Shamir paradigm is applied to 3-round identification schemes, or more generally to what are called Σ-protocols. All these protocols are special honest-verifier zero-knowledge (see Definition 10). Therefore, Theorem 4 and Corollary 2 imply (black-box) negative results for the Fiat-Shamir paradigm

when applied to any such protocol. In what follows we give two specific examples, keeping in mind that there are many other natural examples that we do not mention.

Perfect Zero-Knowledge Protocol for Quadratic Residuosity. Recall the language L_{QR} of quadratic residues.

$$L_{\mathsf{QR}} = \{(N, y) \mid \exists x \in \mathbb{Z}_N^* \text{ s.t. } y = x^2 \mod N\}$$

This language is assumed to be hard w.r.t. distributions $\mathcal{X}_n$ and $\bar{\mathcal{X}}_n$, defined as follows. In both distributions, N is sampled by sampling two random n-bit primes p and q, and setting $N = pq$; in $\mathcal{X}_n$, the element y is a random quadratic residue, and in $\bar{\mathcal{X}}_n$ the element y is a random quadratic non-residue with Jacobi symbol 1.

Recall the well-known *perfect* zero-knowledge Σ-protocol for quadratic residuosity with soundness $1/2$ [Blu81]. We denote by $\Pi^{\ell\text{-}\mathsf{QR}}$ the perfect special honest-verifier zero-knowledge protocol consisting of ℓ parallel executions of the basic Σ-protocol. We denote by $\Pi^{\mathsf{FS}(\ell\text{-}\mathsf{QR})}$ the protocol obtained when applying the Fiat-Shamir paradigm to $\Pi^{\ell\text{-}\mathsf{QR}}$. By applying Corollary 2, we obtain the following theorem:

Theorem 5. *For any $\ell = \ell(n) = \mathrm{poly}(n)$, if L_{QR} is sub-exponentially hard then the soundness of $\Pi^{\mathsf{FS}(\ell\text{-}\mathsf{QR})}$ cannot be proven via a black-box reduction to a falsifiable assumption (unless the assumption is false).*

Blum's Zero-Knowledge Protocol for NP. Recall the well-known Σ-protocol for NP of Blum [Blu87], based on the NP-complete problem of Graph Hamiltonicity, with soundness $1/2$. We denote by $\Pi^{\ell\text{-}\mathsf{Blum}}$ the special honest-verifier zero-knowledge protocol consisting of ℓ parallel executions of the basic Σ-protocol. Note that $\Pi^{\ell\text{-}\mathsf{Blum}}$ is special honest-verifier 2^ℓ-zero-knowledge, if the hiding property of the commitment scheme holds against 2^ℓ-size adversaries.[3]

We denote by $\Pi^{\mathsf{FS}(\ell\text{-}\mathsf{Blum})}$ the protocol obtained when applying the Fiat-Shamir paradigm to $\Pi^{\ell\text{-}\mathsf{Blum}}$. By applying Corollary 2, we obtain the following theorem:

Theorem 6. *For any $\ell = \ell(n) = \mathrm{poly}(n)$, if there exist NP languages L which are sub-exponentially hard, and if $\Pi^{\mathsf{FS}(\ell\text{-}\mathsf{Blum})}$ is instantiated with a commitment scheme whose hiding property holds against 2^ℓ-size adversaries, then the soundness of $\Pi^{\mathsf{FS}(\ell\text{-}\mathsf{Blum})}$ cannot be proven via a black-box reduction to a falsifiable assumption (unless the assumption is false).*

As noted above, one can apply Theorem 4 or Corollary 2 to many other Σ protocols (such as the ones based on the DDH assumption or on the N'th residuosity assumption), and obtain (black-box) negative results for the soundness of the resulting protocols obtained by applying the Fiat-Shamir paradigm.

[3] Recall that for a protocol to be special honest-verifier 2^ℓ-zero knowledge, the simulated view needs to be 2^ℓ-indistinguishable from the real view (see Definition 10).

Proof Intuition for Theorem 4. Theorem 4 follows from the following lemma and from Theorem 3:

Lemma 1. *Let Π be a 3-round public-coin proof or argument system for a $T(n)$-hard language L with the following properties:*

- *The length of the second message, β, satisfies $2^{|\beta|} \leq T$.*
- *Π is special honest verifier (auxiliary input) T-zero knowledge.*

Then the FS-collapse of Π, namely, Π^{FS} is (auxiliary-input) T-zero-knowledge.

Proof Idea. In order to show that Π^{FS} is (auxiliary-input) T-zero knowledge, we must present a simulator $\mathcal{S}^{\mathsf{FS}}$ that simulates the view of every poly-sized circuit $\mathcal{V}^*$. Informally, $\mathcal{S}^{\mathsf{FS}}$ does the following:

- Begin an emulation of $\mathcal{V}^*$ and continue until $\mathcal{V}^*$ outputs h^{FS}.
- Choose T^2 random values $\beta_1, \ldots, \beta_{T^2}$
- Invoke $\mathcal{S}$, the special honest verifier T-zero-knowledge simulator for Π, T^2 times on $\beta_1, \ldots, \beta_{T^2}$, receiving transcripts $(\alpha_1, \beta_1, \gamma_1), \ldots, (\alpha_{T^2}, \beta_{T^2}, \gamma_{T^2})$.
- Return the first transcript $(\alpha_i, \beta_i, \gamma_i)$, such that $h^{\mathsf{FS}}(\alpha_i) = \beta_i$. If no such transcript exists, return $\perp$.

We show that if there is a distinguisher D of size poly$(T(n))$ that can distinguish between real and simulated transcripts outputted by $\mathcal{S}^{\mathsf{FS}}$, then there is also a distinguisher D^* of size poly$(T(n))$ that distinguishes between sequences of length T^2 of real and simulated transcripts ouputted by $\mathcal{S}$. This contradicts the special honest verifier (auxiliary-input) T-zero knowledge of Π.

Intuitively, D^* will emulate $\mathcal{S}^{\mathsf{FS}}$, but will receive transcripts $(\alpha_i, \beta_i, \gamma_i)$, from an external challenger which are either sampled from the real distribution or which are sampled from $\mathcal{S}$. Then, D^* will run D on the view outputted by the emulation and will output whatever D outputs.

Now, in the case that $(\alpha_i, \beta_i, \gamma_i)$, are sampled from the real distribution, $\mathcal{S}^{\mathsf{FS}}$ outputs $\perp$ with negligible (in T) probability. This is the case since in the real distribution, each β_i is independent of α_i and so the probability that $h^{\mathsf{FS}}(\alpha_i) \neq \beta_i$ is $1 - 1/T$. Therefore, the probability that for all $1 \leq i \leq T^2$, $h^{\mathsf{FS}}(\alpha_i) \neq \beta_i$, is at most $(1-1/T)^{T^2}$. Thus when $(\alpha_i, \beta_i, \gamma_i)$, are sampled from the real distribution, the output of D^* is statistically close to a real execution of Π^{FS}.

On the other hand, when $(\alpha_i, \beta_i, \gamma_i)$ are outputted by $\mathcal{S}$, then the output of D^* is identical to the output of $\mathcal{S}^{\mathsf{FS}}$. Thus, D, and so also D^*, will distinguish between the two cases.

4.3 Separating CS Proofs from Falsifiable Assumptions

In this section we show that for sufficiently hard NP languages, there exist probabilistically checkable proofs (PCPs) such that Micali's CS proofs [Mic94] instantiated with such a PCP cannot be proven sound (even when the statement is chosen "non-adaptively") via a black-box reduction to any falsifiable assumption.

Let Π^{FS} denote Micali's 2-message CS proof system obtained by applying the Fiat-Shamir transformation to Kilian's succinct argument system [Kil92] Π using $h_s \leftarrow \mathcal{H}$. For any NP language L and any PCP, Π_{pcp}, for L, Micali proved that Π^{FS} is sound in the so called *random oracle model*, where the FS-hash h_s is modeled as a random oracle. We now prove that for every $2^{\ell(n)}$-hard language L, there exists an ℓ-query PCP such that the CS proof Π^{FS} for language L cannot be proven sound via a black-box reduction to any falsifiable assumption. More formally,

Theorem 7. *For all $\ell = \ell(n)$ and any $2^{\ell(n)}$-hard language L, there exists an ℓ-query PCP Π_{pcp} such that the soundness of CS proof Π^{FS} instantiated with Π_{pcp} for language L cannot be proven via a black-box reduction to a $2^{\ell(n)}$-hard falsifiable assumption (unless the assumption is false).*

The following corollary follows easily from Theorem 7.

Corollary 3. *For any sub-exponentially hard language L and for any $\ell = \mathrm{poly}(n)$, there exists an ℓ-query PCP Π_{pcp} such that the soundness of CS proof Π^{FS} instantiated with Π_{pcp} for language L cannot be proven via a black-box reduction to a $2^{\ell(n)}$-hard falsifiable assumption (unless the assumption is false).*

Let L be a $2^{\ell(n)}$-hard language. Our main idea is to show that when Kilian's succinct argument Π is instantiated with a specific PCP (with some zero-knowledge properties), then it is a (special) honest verifier $2^{\ell(n)}$-zero knowledge argument for L, where the verifier's second message is of length at most ℓ. This, when combined with Theorem 4 immediately yields the proof of Theorem 7. Due to lack of space, we defer the proof to the full version [DJKL12].

References

[AABN02] Abdalla, M., An, J.H., Bellare, M., Namprempre, C.: From Identification to Signatures via the Fiat-Shamir Transform: Minimizing Assumptions for Security and Forward-Security. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 418–433. Springer, Heidelberg (2002)

[BGW12] Bitansky, N., Garg, S., Wichs, D.: Why "fiat-shamir for proofs" lacks a proof. Cryptology ePrint Archive, Report 2012/705 (2012), `http://eprint.iacr.org/`

[Blu81] Blum, M.: Coin flipping by telephone. In: Proceedings of the 18th Annual International Cryptology Conference, CRYPTO 1981, pp. 11–15 (1981)

[Blu87] Blum, M.: How to prove a theorem so no one else can claim it. In: Proceedings of the International Congress of Mathematicians, pp. 1444–1451 (1987)

[BLV03] Barak, B., Lindell, Y., Vadhan, S.P.: Lower bounds for non-black-box zero knowledge. In: 44th Annual Symposium on Foundations of Computer Science, pp. 384–393. IEEE Computer Society Press (October 2003)

[BR93] Bellare, M., Rogaway, P.: Random oracles are practical: A paradigm for designing efficient protocols. In: Ashby, V. (ed.) ACM CCS 1993: 1st Conference on Computer and Communications Security, pp. 62–73. ACM Press (November 1993)

[Bro05] Brown, D.R.L.: Breaking rsa may be as difficult as factoring. Cryptology ePrint Archive, Report 2005/380 (2005), http://eprint.iacr.org/

[BV98] Boneh, D., Venkatesan, R.: Breaking RSA May Not Be Equivalent to Factoring. In: Nyberg, K. (ed.) EUROCRYPT 1998. LNCS, vol. 1403, pp. 59–71. Springer, Heidelberg (1998)

[Cor02] Coron, J.-S.: Security Proof for Partial-Domain Hash Signature Schemes. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 613–626. Springer, Heidelberg (2002)

[Cra12] Cramer, R. (ed.): TCC 2012. LNCS, vol. 7194. Springer, Heidelberg (2012)

[DHT12] Dodis, Y., Haitner, I., Tentes, A.: On the instantiability of hash-and-sign rsa signatures. In: Cramer [Cra12], pp. 112–132

[DJKL12] Dachman-Soled, D., Jain, A., Kalai, Y.T., Lopez-Alt, A.: On the (in)security of the fiat-shamir paradigm, revisited. Cryptology ePrint Archive, Report 2012/706 (2012), http://eprint.iacr.org/

[DNRS99] Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.J.: Magic functions. In: FOCS, pp. 523–534 (1999)

[DOP05] Dodis, Y., Oliveira, R., Pietrzak, K.: On the Generic Insecurity of the Full Domain Hash. In: Shoup, V. (ed.) CRYPTO 2005. LNCS, vol. 3621, pp. 449–466. Springer, Heidelberg (2005)

[DRV12] Dodis, Y., Ristenpart, T., Vadhan, S.P.: Randomness condensers for efficiently samplable, seed-dependent sources. In: Cramer [Cra12], pp. 618–635

[FS86] Fiat, A., Shamir, A.: How to Prove Yourself: Practical Solutions to Identification and Signature Problems. In: Odlyzko, A.M. (ed.) CRYPTO 1986. LNCS, vol. 263, pp. 186–194. Springer, Heidelberg (1987)

[GBL08] Garg, S., Bhaskar, R., Lokam, S.V.: Improved Bounds on Security Reductions for Discrete Log Based Signatures. In: Wagner, D. (ed.) CRYPTO 2008. LNCS, vol. 5157, pp. 93–107. Springer, Heidelberg (2008)

[GK03] Goldwasser, S., Kalai, Y.T.: On the (in)security of the Fiat-Shamir paradigm. In: 44th Annual Symposium on Foundations of Computer Science, pp. 102–115. IEEE Computer Society Press (October 2003)

[GMR89] Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof systems. SIAM Journal on Computing 18(1), 186–208 (1989); Preliminary version appeared in STOC 1985.

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

[GQ90] Guillou, L.C., Quisquater, J.-J.: A "Paradoxical" Identity-Based Signature Scheme Resulting from Zero-Knowledge. In: Goldwasser, S. (ed.) CRYPTO 1988. LNCS, vol. 403, pp. 216–231. Springer, Heidelberg (1990)

[GW11] Gentry, C., Wichs, D.: Separating succinct non-interactive arguments from all falsifiable assumptions. In: Fortnow, L., Vadhan, S.P. (eds.) 43rd Annual ACM Symposium on Theory of Computing, pp. 99–108. ACM Press (June 2011)

[HH09] Haitner, I., Holenstein, T.: On the (Im)Possibility of Key Dependent Encryption. In: Reingold, O. (ed.) TCC 2009. LNCS, vol. 5444, pp. 202–219. Springer, Heidelberg (2009)

[Kil92] Kilian, J.: A note on efficient zero-knowledge proofs and arguments. In: Proceedings of the 24th Annual ACM Symposium on Theory of Computing, STOC 1992, pp. 723–732 (1992)

[Mic94] Micali, S.: A secure and efficient digital signature algorithm. Technical Memo MIT/LCS/TM-501b, Massachusetts Institute of Technology, Laboratory for Computer Science (April 1994)

[Mic00] Micali, S.: Computationally sound proofs. SIAM Journal on Computing 30(4), 1253–1298 (2000); Preliminary version appeared in FOCS 1994

[Nao03] Naor, M.: On Cryptographic Assumptions and Challenges. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 96–109. Springer, Heidelberg (2003)

[Oka93] Okamoto, T.: Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes. In: Brickell, E.F. (ed.) CRYPTO 1992. LNCS, vol. 740, pp. 31–53. Springer, Heidelberg (1993)

[Pas03] Pass, R.: Simulation in Quasi-Polynomial Time, and Its Application to Protocol Composition. In: Biham, E. (ed.) EUROCRYPT 2003. LNCS, vol. 2656, pp. 160–176. Springer, Heidelberg (2003)

[Pas11] Pass, R.: Limits of provable security from standard assumptions. In: Fortnow, L., Vadhan, S.P. (eds.) 43rd Annual ACM Symposium on Theory of Computing, pp. 109–118. ACM Press (June 2011)

[PS00] Pointcheval, D., Stern, J.: Security arguments for digital signatures and blind signatures. Journal of Cryptology 13(3), 361–396 (2000)

[PV05] Paillier, P., Vergnaud, D.: Discrete-Log-Based Signatures May Not Be Equivalent to Discrete Log. In: Roy, B. (ed.) ASIACRYPT 2005. LNCS, vol. 3788, pp. 1–20. Springer, Heidelberg (2005)

[Sch91] Schnorr, C.-P.: Efficient signature generation by smart cards. Journal of Cryptology 4(3), 161–174 (1991)

[Seu12] Seurin, Y.: On the Exact Security of Schnorr-Type Signatures in the Random Oracle Model. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 554–571. Springer, Heidelberg (2012)

[Wic12] Wichs, D.: Barriers in cryptography with weak, correlated and leaky sources. Cryptology ePrint Archive, Report 2012/459 (2012), `http://eprint.iacr.org/`