From Obfuscation to the Security of Fiat-Shamir for Proofs

Yael Tauman Kalai* Guy N. Rothblum† Ron D. Rothblum‡

March 14, 2016

Abstract

The Fiat-Shamir paradigm [CRYPTO'86] is a heuristic for converting three-round identification schemes into signature schemes, and more generally, for collapsing rounds in constant-round public-coin interactive protocols. This heuristic is very popular both in theory and in practice, and its security has been the focus of extensive study.

In particular, this paradigm was shown to be secure in the so-called Random Oracle Model. However, in the plain model, mainly negative results were shown. In particular, this heuristic was shown to be *insecure* when applied to *computationally sound* proofs (also known as arguments). Moreover, recently it was shown that even in the restricted setting, where the heuristic is applied to interactive *proofs* (as opposed to arguments), its soundness cannot be proven via a black-box reduction to any so-called *falsifiable* assumption.

In this work, we give a *positive result* for the security of this paradigm in the *plain model*. Specifically, we construct a hash function for which the Fiat Shamir paradigm is *secure* when applied to proofs (as opposed to arguments), assuming the existence of a sub-exponentially secure indistinguishability obfuscator, the existence of an exponentially secure input-hiding obfuscator for the class of multi-bit point functions, and the existence of a sub-exponentially secure one-way function.

1 Introduction

In 1986, Fiat and Shamir [FS86] proposed a general method for reducing interaction in any constant-round public-coin protocol by replacing the verifier with a hash function. Initially, this heuristic was proposed for the sake of transforming three-round public-coin identification (ID) schemes into digital signature schemes. This so-called *Fiat-Shamir heuristic*, quickly gained popularity both in theory and in practice, since known ID schemes (in which a sender *interactively* identifies himself to a receiver) are significantly simpler and more efficient than known signature schemes, and thus this heuristic gives an efficient and easy way to implement digital signature schemes.

The Fiat-Shamir heuristic also has important applications outside the regime of ID and signature schemes. For example, it was used by Micali in his construction of CS-proofs [Mic94]. More generally, the importance of the Fiat-Shamir heuristic stems from the fact that latency, caused by sending messages back and forth, is often a bottleneck in running cryptographic protocols [MNPS04, BDNP08].

*Microsoft Research. Email: yael@microsoft.com.

†Samsung Research America. Email: rothblum@alum.mit.edu.

‡MIT. Email: ronr@mit.edu. Supported by NSF Frontier "TWC: TTP Option: Frontier: Collaborative: MACS: A Modular Approach to Cloud Security" - CNS1413920 Simons Foundation - Agreement Dated 6-5-12.

The Fiat-Shamir method is extremely simple and intuitive: The basic idea is to reduce interaction by having the verifier send the prover a hash function H (chosen at random from some family of hash functions). The prover then "simulates" all the verifier's messages on his own by applying H to the transcript so far. For example, a three-message interactive proof, where we denote the transcript by (α, β, γ), is converted to the following 2-message protocol, where the verifier first sends a hash function H to the prover, and then the prover simulates the three messages on his own as follows: He first computes his first message α, then he computes the verifier's message β by setting $\beta = H(\alpha)$, and finally he computes his final message γ, and sends (α, β, γ) to the verifier.

The intuition for why this method is secure, is that if H looks like a truly random function, and if all the prover can do is use H in a black-box manner, then interacting with H is similar to interacting with the real verifier, and hence security follows. This intuition was formalized by Pointcheval and Stern [PS96], who proved that the Fiat-Shamir heuristic is secure in the so-called *Random Oracle Model* (ROM) – when the hash function is modeled by a random oracle [BR93]. This led to the belief that if a 2-message protocol, obtained by applying the Fiat-Shamir paradigm, is insecure, then it must be the case that the hash family used is not "secure enough", and the hope was that there exists another hash family that is sufficiently secure.

Since Pointcheval and Stern published their positive result (in the ROM), and due to the popularity and importance of the Fiat-Shamir heuristic, many researchers tried to prove the security of this paradigm in the plain model. Unfortunately, these attempts led mainly to negative results. Barak [Bar01] gave the first negative result, by constructing a (contrived) constant-round public-coin protocol such that when the Fiat-Shamir heuristic is applied to it, the resulting 2-round protocol is not sound, no matter which hash family is used. In a followup work, Goldwasser and Kalai [GK03], gave another (contrived) construction for a 3-round public-coin ID scheme, for which the resulting signature scheme obtained by applying the Fiat-Shamir heuristic, is insecure, no matter which hash family is used. However, both these negative results are for protocols that are only *computationally sound*, also known as *arguments*.

This gave rise to the following question:

Is the Fiat-Shamir method secure when applied to interactive proofs (as opposed to arguments)?

Barak, Lindell and Vadhan [BLV03] presented a security property for the Fiat-Shamir hash function, which if realized, would imply the security of the Fiat-Shamir paradigm applied to any constant-round public-coin interactive proof system.[1] However, they left open the problem of realizing this security definition under standard hardness assumptions (or under any assumption beyond simply assuming that the definition holds for a given hash function). Recently, Dodis, Ristenpart and Vadhan [DRV12] showed that under specific assumptions regarding the existence of robust randomness condensers for seed-dependent sources, the definitions of [BLV03] can be realized. However, the question of constructing such suitable robust randomness condensers was left open by [DRV12].

On the other hand, Bitansky *et. al.* [BDG$^+$13] gave a negative result. They showed that that soundness of the Fiat-Shamir paradigm, even when applied to interactive proofs, cannot be proved via a black-box reduction to any so-called *falsifiable* assumption (see Naor [Nao03]).[2]

[1] Loosely speaking, a hash family $\{h_s\}$ is said to have this security property if for every probabilistic polynomial time adversary $\mathcal{A}$, that is given a random seed s and outputs an element in the domain of h_s, the random variable $h_s(\mathcal{A}(s))$ conditioned on $\mathcal{A}(s)$ has almost full min entropy.

[2] Our assumptions (see Section 1.1), which deal with exponential-time (rather than polynomial-time) adversaries,

Finally, we remark that in a recent work Canetti, Chen and Reyzin [CCR15] construct a *correlation intractable* function ensemble that withstands relations that can be computed in a-priori bounded polynomial complexity. This does not have implications to the security of the Fiat-Shamir paradigm, where we need correlation intractable ensembles for hard-to-compute relations. A further discussion follows the description of our results.

1.1 Our Results

In this work, we prove that the Fiat-Shamir paradigm when applied to interactive proofs (as opposed to arguments) is *sound*, under the following three cryptographic assumptions:

1. The existence of 2^n-secure indistinguishability obfuscation iO, where 2^n is the domain size of the functions being obfuscated.[3]

   Recently, several constructions of iO obfuscation were proposed, starting with the work of Garg *et al.* [GGH$^+$13]. However, to date, non of these constructions are known to be provably secure under what is known as a complexity assumption [GK16] or more generally a falsifiable assumption [Nao03]. We mention that [GLSW14] provided a construction and proved its security under the subgroup elimination assumption, which is a complexity assumption (and in particular is a falsifiable assumption). However, this assumption has been refuted in all candidate multi-linear groups.

2. The existence of 2^n-secure puncturable pseudo-random function (PRF) family $\mathcal{F}$, where 2^n is the domain size.

   Puncturable PRFs were defined in [BW13, BGI14, KPTZ13]. The PRF family of [GGM86] is a puncturable PRF family, and thus 2^n-secure puncturable PRFs can be constructed from any sub-exponentially secure one-way function.

3. The existence of an exponentially secure input-hiding obfuscation hideO for the class of multi-bit point functions $\{\mathcal{I}_{n,k}\}$. The class $\{\mathcal{I}_{n,k}\}$ consists of functions of the form $I_{\alpha,\beta}$ where $|\alpha| = n$ and $|\beta| = k$, and where $I_{\alpha,\beta}(x) = \beta$ for $x = \alpha$ and $I_{\alpha,\beta}(x) = 0$ otherwise. An obfuscation for this class is said to be input-hiding with T-security if any *poly-size* adversary that is given an obfuscation of a random function $I_{\alpha,\beta}$ in this family, guesses α with probability at most T^{-1}. We note that the value β may be correlated with α and furthermore, it may be computationally difficult to find β from α. For our construction we require T which is roughly equal to $2^n/\mu$, where μ is the soundness of the underlying proof-system. For example, if we start off with an interactive-proof with soundness 2^{-n^ϵ}, then we require roughly $T = 2^{n-n^\epsilon}$.

   This assumption was considered in [CD08, BC14], who also provided a candidate construction based on a strong variant of the DDH assumption (we elaborate on this in Section 2.4).[4]

Theorem 1.1. *[(Informally Stated)] Under the assumptions above, for any constant-round interactive proof* Π*, the resulting* 2*-message argument* Π^{FS}*, obtained by applying the Fiat-Shamir paradigm to* Π *with the function family* $\mathsf{iO}(\mathcal{F})$*, is secure.*

are inherently not falsifiable.

[3] This assumption has been made in many previous works on iO and is referred to as sub-exponential iO since the security parameter can be polynomially larger than n (which makes 2^n sub-exponential in the security parameter).

[4] While DDH (and even discrete log) can be broken in time less than 2^n (even in the generic group model - e.g., by the baby-step giant-step algorithm), this does not imply a non-trivial *polynomial-time* attack (i.e., one with success probability greater than $\mathrm{poly}(n)/2^n$).

Here and throughout this work $\mathsf{iO}(\mathcal{F})$ refers to an iO obfuscation of a program that computes the PRF, using a hardwired random seed.

This result has interesting corollaries. In particular, Dwork *et. al.* [DNRS99] (and independently, Hada and Tanaka [HT98]) observed an intriguing connection between the security of the Fiat-Shamir paradigm and the existence of certain zero-knowledge protocols. In particular, if there exists a constant-round public-coin zero-knowledge proof for a language outside BPP, then the Fiat-Shamir paradigm is not secure when applied to this zero-knowledge proof. Intuitively, this follows from the following observation: Consider the cheating verifier that behaves exactly like the Fiat-Shamir hash function. The fact that the protocol is zero-knowledge implies that there exists a simulator who can simulate the view in an indistinguishable manner. Thus, for elements in the language the simulator generates accepting transcripts. The simulator cannot distinguish between elements in the language and elements outside the language (since the simulator runs in poly-time and the language is outside of BPP). In addition, the protocol is public coin, which implies that the simulator knows whether the transcript is accepted or not. Hence, it must be the case that the simulator also generates accepting transcripts for elements that are not in the language, which implies that the Fiat-Shamir paradigm is not secure.

Thus, Theorem 3.1 implies the following corollary.

Corollary 1.2. *Under the assumptions above, there does not exist a constant-round public-coin zero-knowledge proof with negligible soundness for languages outside* BPP*.*

In particular, this corollary implies that (under the assumptions above) parallel repetition of Blum's Hamiltonicity protocol for NP [Blu87] is not zero-knowledge. Previously it was not known whether (in general) parallel repetition preserves zero-knowledge. Our result shows that it does not (under the assumptions above).

We note that even for those who are skeptical about the obfuscation assumptions we make, this corollary implies that finding a constant-round public-coin zero-knowledge proof requires overcoming technical barriers, and in particular requires disproving the existence of sub-exponentially secure iO obfuscation, or the existence of exponentially secure input-hiding obfuscation for the class of multi-bit point functions (or, less likely, disproving the existence of sub-exponential OWF).

Comparison to Canetti *et al.* [CCR15]. As mentioned above, in very recent work [CCR15] construct a correlation intractable function ensemble that withstands all relations computable in a-priori bounded polynomial complexity. Namely, for any fixed polynomial p, they construct a function ensemble as follows: for any evasive (see below) relation R computable in time p, given a random function f in the ensemble, it is hard to find x such that $(x, f(x)) \in R$.

As mentioned above, this result does not have any implications to the security of the Fiat-Shamir paradigm, since to prove the security of this paradigm we need a correlation intractable ensemble for relations that cannot be computed in polynomial time.

In terms of the assumptions used, [CCR15] assume the existence of sub-exponentially secure indistinguishability obfuscation, the existence of a sub-exponentially secure puncturable PRF family, and the existence of input-hiding obfuscation for the class of evasive functions. An evasive family is a collection of functions where for any input x, a random function from the collection outputs 0 on x with overwhelming probability [BBC+14]. Comparing to the assumptions we make in this work, we also make the first two assumptions. However, we assume input-hiding obfuscation only for multi-bit point functions (a significantly smaller family compared to general evasive functions).

On the other hand, we require an exponentially secure input-hiding obfuscation, whereas their work only needs polynomial-time hardness of the input-hiding obfuscation.

1.2 Overview

Throughout this overview we focus on proving the security of the Fiat-Shamir paradigm, when applied to 3-round public-coin interactive proofs. The more general case, of any constant number[5] of rounds, is then proved by induction on the number of rounds (we refer the reader to Section 4 for details). Consider any 3-round proof Π for a language L. Denote the transcript by (α, β, γ) where α is the first message sent by the prover, β is the random message sent by the verifier, and γ is the final message sent by the prover. Fix any $x \notin L$. The fact that Π is a sound proof means that for every α, for most of the verifier's messages β, there does not exist γ that makes the verifier accept.

The basic idea stems from the original intuition for why the Fiat-Shamir is secure, which is that if we use a hash function H that looks like a truly random function, then all the prover can do is use H in a black-box manner, in which case interacting with H is similar to interacting with the real verifier, and hence security follows.

The first idea that comes to mind is to choose the hash function randomly from a pseudo-random function (PRF) family. However, the security guarantee of a PRF is that given only *black-box* access to a random function f in the PRF family, one cannot distinguish it from a truly random function. No guarantees are given if the adversary is given a succinct circuit for computing f.

Obfuscation to the Rescue. A natural next step is to try to obfuscate f, in the hope that whatever can be learned given the obfuscation of f can also be learned from black-box access to f. However, this requires virtual-black-box (VBB) security, and VBB obfuscation is known not to exist [BGI+12]. Moreover, there are specific PRF families for which VBB obfuscation is impossible [BGI+12]. Further obstacles to VBB obfuscation of PRFs and, more generally, functions with high pseudo-entropy (w.r.t. auxiliary input) are given in [GK05, BCC+14]. Given these obstacles to achieving VBB obfuscation, could we hope to prove security using relaxed notions of obfuscation, such as iO obfuscation? The question is:

Is iO *obfuscation strong enough to prove the security of the Fiat-Shamir paradigm?*

It is well known that iO obfuscation is *not* strong enough to prove the security of the Fiat-Shamir paradigm when applied to computationally sound interactive *arguments*. Indeed the Fiat-Shamir paradigm is known be insecure when applied to arguments as opposed to proofs.[6] In contrast, we show that iO obfuscation (together with additional assumptions) is strong enough to prove security when the Fiat-Shamir paradigm is applied to interactive *proofs* (rather than arguments).

For proving security of the Fiat-Shamir paradigm for *proofs*, consider a cheating prover for the transformed protocol Π^{FS}, who receives the obfuscation $\mathsf{iO}(f_s)$ of a pseudo-random function f_s. Since f_s is a PRF, we know that there will only be a small set Bad_s of inputs α (corresponding to the prover's first message in the proof Π), for which the communication prefix $(\alpha, f_s(\alpha))$ can lead

[5] The Fiat Shamir paradigm refers to constant round protocols. Indeed, there are interactive proofs with a super-constant number of rounds (and negligible soundness error) for which the Fiat Shamir paradigm is insecure.

[6] More specifically, the insecurity is in the sense that there exist contrived interactive arguments such that for any hash family $\mathcal{H}$, applying the Fiat-Shamir paradigm with the hash family $\mathcal{H}$, results in an insecure 2-round protocol [Bar01, GK03].

the verifier in the interactive proof to accept (i.e. α's for which there exists γ s.t. $(\alpha, f(\alpha), \gamma)$ is an accepting transcript).

To show the security of the resulting protocol, we now want to claim that the obfuscation *hides* this (small) set Bad_s of inputs, and that a cheating prover P^* cannot find any input $\alpha \in \mathsf{Bad}_s$. Note, however, that iO obfuscation only guarantees that one cannot distinguish between the obfuscation of two functionally equivalent circuits of the same size, and it does not give any hiding guarantees.

Puncturable PRFs to the Rescue? As mentioned above, iO obfuscation does not immediately seem to give any hiding guarantees. Nonetheless, starting with the beautiful work of Sahai and Waters [SW14], iO has proved remarkably powerful in the construction of a huge variety of cryptographic primitives. A basic technique used in order to get a hiding guarantee from iO obfuscation, as pioneered in [SW14], is to use it with a puncturable PRF family.

A puncturable PRF family is a PRF family that allows the "puncturing" of the seed at any point α in the domain of f. Namely, for any point α in the domain, and for any seed s of the PRF, one can generate a "punctured" seed, denoted by $s\{\alpha\}$. This seed allows the computation of f_s anywhere in the domain, except at point α, with the security guarantee that for a random seed s chosen independently of α, the element $f_s(\alpha)$ looks (computationally) random given $(s\{\alpha\}, \alpha)$. The security of iO obfuscation guarantees that one cannot distinguish between $\mathsf{iO}(s)$ and $\mathsf{iO}(s\{\alpha\}, \alpha, f_s(\alpha))$,[7] which together with the security of the puncturable PRF, implies that one cannot distinguish between $\mathsf{iO}(s)$ and $\mathsf{iO}(s\{\alpha\}, \alpha, u)$ for a truly random output u. Thus, we managed to use iO, together with the puncturing technique, to generate a circuit for computing f_s that hides the value of $f_s(\alpha)$. We emphasize that this technique crucially relies on the fact that the punctured point α is independent of the seed s, and hence as a result $f_s(\alpha)$ is computationally random.

It is natural to try and use obfuscated puncturable PRFs to show security of the Fiat-Shamir paradigm. Consider the following naive (and flawed) analysis, which loosely speaking proceeds in three steps: Suppose that there exists a poly-size cheating prover P^* that convinces the verifier to accept $x \notin L$. Recall that we denote transcripts by (α, β, γ). The (statistical) soundness of Π implies that for every α, for most of the verifier's messages β, there does not exist γ that makes the verifier accept. For any function f consider the (evasive) relation $R = \{(\alpha, \beta) : \exists \gamma \text{ s.t. } V(x, \alpha, \beta, \gamma) = 1\}$. Suppose that the cheating prover P^*, given $\mathsf{iO}(s)$, outputs α such that $(\alpha, f_s(\alpha)) \in R$, with non-negligible probability.

1. Puncture the PRF at a random point α^* s.t. $\alpha^* \in \mathsf{Bad}_s$, and send the obfuscation of $\mathsf{iO}(s\{\alpha^*\}, \alpha^*, f_s(\alpha^*))$ to the cheating prover P^*. Note that this does not change the functionality.

   Therefore, we can use the (sub-exponential) security of iO to argue that the cheating prover P^* cannot tell where we punctured the PRF, and still succeeds with non-negligible probability. In particular, taking M to be the expected number of α's such that $(\alpha, f_s(\alpha)) \in R$, we have that P^* outputs α^* with probability $\approx 1/M$ (up to $\mathrm{poly}(n)$ factors).[8]

2. Next, we want to use the (sub-exponential) security of the puncturable PRF to argue that the cheating prover P^* cannot distinguish between $(s\{\alpha^*\}, \alpha^*, f_s(\alpha^*))$ and $(s\{\alpha^*\}, \alpha^*, \beta^*)$ where

[7] We use $(s\{\alpha\}, \alpha, f_s(\alpha))$ to denote the circuit that on input α outputs the hardwired value $f_s(\alpha)$, and on any other input $x \neq \alpha$ computes $f_s(x)$ using the punctured seed $s\{\alpha\}$.

[8] We think of n as polynomially related to the security parameter, where 2^n is the domain size of f_s.

(α^*, β^*) is random in R. Thus, given $\mathsf{iO}(s\{\alpha^*\}, \alpha^*, \beta^*)$ the cheating prover P^* still outputs α^* with probability $\approx 1/M$ (up to $\text{poly}(n)$ factors).

3. In the final step, we argue that α^* is close to uniform (for an appropriate modification of the original protocol) and independent of s. Thus, given $\mathsf{iO}(s\{\alpha^*\}, \alpha^*, \beta^*)$, the cheating prover P^* outputs α^* with probability $\approx 1/M$ (up to $\text{poly}(n)$ factors), where α^* is close to truly random. We want to argue that this contradicts the (sub-exponential) security of iO.

Unfortunately, the argument sketched above is doubly-flawed. In particular, the arguments in Step (2) and Step (3) are simply false. In Step (2) we start with a distribution where f_s is punctured at a point α^* for which $(\alpha^*, f_s(\alpha^*))$ is not (computationally) random, and in fact *the choice of* α^* *depends on the seed* s. We want to argue that this is indistinguishable from the case where we pick (α^*, β^*) randomly in R, and then puncture at α^*. It is not a-priori clear why the puncturable PRF or iO would guarantee this indistinguishability. Indeed, the functions generated by these two distributions can be distinguished with some advantage by simply counting the number of input-output pairs that are in R.

Nevertheless, in our analysis (see Lemma 3.3) we manage to argue that the cheating prover P^*, given $\mathsf{iO}(s\{\alpha^*\}, \alpha^*, \beta^*)$ where (α^*, β^*) is random in R, still outputs α^* with probability significantly higher than $1/2^n$ (i.e., significantly higher than guessing). Indeed, P^* still outputs α^* with probability $\approx 1/M$ (up to $\text{poly}(n)$ factors).

We next move to the flaw in Step (3). The problem here is that puncturing at the point α^* *does not at all hide* α^*. It is also not clear whether the iO obfuscation of the punctured seed hides α^*.

Input-Hiding Obfuscation to the Rescue. We overcome this hurdle by using an exponentially secure input-hiding obfuscation to hide the punctured point.

Namely, we replace $\mathsf{iO}(s\{\alpha^*\}, \alpha^*, \beta^*)$ with $\mathsf{iO}(s, \mathsf{hideO}(\alpha^*, \beta^*))$, where hideO is an exponentially secure input hiding obfuscator, and where we did not change the functionality of the circuit; i.e. the circuit on input x first runs $\mathsf{hideO}(\alpha^*, \beta^*)$ to check if $x = \alpha^*$; if so it outputs β^* and otherwise it outputs $f_s(x)$. The security of iO implies that $P^*(\mathsf{iO}(s, \mathsf{hideO}(\alpha^*, \beta^*)))$ outputs α^* with probability $1/M$ (up to $\text{poly}(n)$ factors).

It remains to note that s is independent of (α^*, β^*), and hence we conclude that there exists a poly-size adversary that given $\mathsf{hideO}(\alpha^*, \beta^*)$ outputs α^* with probability $1/M$ (up to $\text{poly}(n)$ factors). In the last step we replace the distribution of (α^*, β^*) with a distribution where α^* is chosen uniformly at random from $\{0,1\}^n$ and β^* is chosen at random such that $(\alpha^*, \beta^*) \in R$ and prove that still there exists a poly-size adversary that given $\mathsf{hideO}(\alpha^*, \beta^*)$ (where (α^*, β^*) is according to the new distribution) outputs α^* with probability $1/M$ (up to $\text{poly}(n)$ factors). This contradicts the exponential security of the input-hiding obfuscator hideO.

Remark 1.3. *We note that the input-hiding obfuscator* was only used in the security analysis. *It plays no role in the construction itself. This is similar to some other recent uses of indistinguishability obfuscation in the literature.*

We hope that the idea of using input-hiding obfuscation to hide the punctured point, will find further applications.

2 Preliminaries

2.1 Indistinguishability.

Definition 2.1. *For any function $T : \mathbb{N} \to \mathbb{N}$ and for any function $\mu : \mathbb{N} \to [0,1]$, we say that $\mu = \text{negl}(T)$ if for every constant $c > 0$ there exists $K \in \mathbb{N}$ such that for every $k \geq K$,*

$$\mu(k) \leq T(k)^{-c}.$$

Definition 2.2. *Two distribution families $\mathcal{X} = \{\mathcal{X}_\kappa\}_{\kappa\in\mathbb{N}}$ and $\mathcal{Y} = \{\mathcal{Y}_\kappa\}_{\kappa\in\mathbb{N}}$ are said to be T-*indistinguishable *(denoted by $\mathcal{X} \overset{T}{\approx} \mathcal{Y}$) if for every circuit family $D = \{D_\kappa\}_{\kappa\in\mathbb{N}}$ of size $\text{poly}(T(\kappa))$,*

$$\text{Adv}_D^{\mathcal{X},\mathcal{Y}}(S) \overset{\mathsf{def}}{=} |\Pr[D(x) = 1] - \Pr[D(y) = 1]| = \text{negl}(T(\kappa)),$$

where the probabilities are over $x \leftarrow \mathcal{X}_\kappa$ and over $y \leftarrow \mathcal{Y}_\kappa$.

2.2 Puncturable PRFs

Our construction uses a 2^n-secure pseudo-random function (PRF) family that is *puncturable* [BW13, BGI14, KPTZ13, SW14], see the definitions below.

Definition 2.3 (T-Secure PRF [GGM86])**.** *Let $m = m(\kappa)$, $n = n(\kappa)$ and $k = k(\kappa)$ be ensembles of integers. A PRF family is an ensemble $\mathcal{F} = \{\mathcal{F}_\kappa\}_{\kappa\in\mathbb{N}}$ of function families, where $\mathcal{F}_\kappa = \{f_s : \{0,1\}^n \to \{0,1\}^k\}_{s\in\{0,1\}^m}$. The PRF $\mathcal{F}$ is T-secure, for $T = T(\kappa)$, if for every $\text{poly}(T)$-size (non-uniform) adversary* Adv*:*

$$\left|\text{Adv}^{f_s}(1^\kappa) - \text{Adv}^{f}(1^\kappa)\right| = \text{negl}(T(\kappa)),$$

where f_s is a random function in $\mathcal{F}_\kappa$, generated using a uniformly random seed $s \in \{0,1\}^{m(\kappa)}$, and f is a truly random function with domain $\{0,1\}^n$ and range $\{0,1\}^k$.

We use 2^n-secure PRF families in our construction (for $k = \text{poly}(n)$). We can construct such PRFs assuming subexponentially hard one-way functions by taking the seed length m to be a sufficiently large polynomial in n. Observe that, since the entire truth table of the function can be constructed in time $\text{poly}(n) \cdot 2^n$, we get that 2^n-security implies that the entire truth table of a PRF f_s is indistinguishable from a uniformly random truth table.[9]

Definition 2.4 (T-Secure Puncturable PRF [SW14])**.** *A T-secure family of PRFs (as in Definition 2.3) is* puncturable *if there exist* PPT *procedures* $\mathsf{puncture}$ *and* eval *such that*

1. *Puncturing a PRF key $s \in \{0,1\}^m$ at a point $r \in \{0,1\}^n$ gives a punctured key $s\{r\}$ that can still be used to evaluate the PRF at any point $r' \neq r$*

$$\forall r \in \{0,1\}^n, r' \neq r : \Pr_{s, s\{r\} \leftarrow \mathsf{puncture}(s,r)} \left[\mathsf{eval}(s\{r\}, r') = f_s(r')\right] = 1$$

[9] The fact that subexponential OWF yield PRFs for which distinguishing the entire truth table from a random truth table the truth table of a random function has been previously noted in the literature, most notably by Razborov and Rudich [RR97] in their work on natural proofs.

2. *For any fixed $r \in \{0,1\}^n$, given a punctured key $s\{r\}$, the value $f_s(r)$ is pseudorandom:*

$$(s\{r\}, r, f_s(r)) \overset{T(\kappa)}{\approx} (s\{r\}, r, u),$$

where $s\{r\}$ is obtained by puncturing a random seed $s \in \{0,1\}^{m(\kappa)}$ at the point r, and u is uniformly random in $\{0,1\}^k$.

We note that the GGM-based construction of PRFs gives a construction of 2^n-secure puncturable PRFs from any subexponentially hard one-way function [GGM86, HILL99].

2.3 Indistinguishability Obfuscation

Our construction uses an indistinguishability obfuscator iO with 2^{-n} security. A candidate construction was first given in the work of Garg *et al.* [GGH+13].

Definition 2.5 (T-secure Indistinguishability Obfuscator [BGI+12])**.** *Let $T : \mathbb{N} \to \mathbb{N}$ be a function. Let $\mathbb{C} = \{\mathbb{C}_n\}_{n\in\mathbb{N}}$ be a family of polynomial-size circuits, where $\mathbb{C}_n$ is a set of boolean circuits operating on inputs of length n. Let iO be a* PPT *algorithm, which takes as input a circuit $C \in \mathbb{C}_n$ and a security parameter $\kappa \in \mathbb{N}$, and outputs a boolean circuit $\mathsf{iO}(C)$ (not necessarily in $\mathbb{C}$).*

iO *is a T-secure* indistinguishability obfuscator *for $\mathbb{C}$ if it satisfies the following properties:*

1. Preserving Functionality: *For every $n, \kappa \in \mathbb{N}$, $C \in \mathbb{C}_n$, $x \in \{0,1\}^n$:*

$$(\mathsf{iO}(C, 1^\kappa))(x) = C(x).$$

2. Indistinguishable Obfuscation: *For every two sequence of circuits $\{C_n^1\}_{n\in\mathbb{N}}$ and $\{C_n^2\}_{n\in\mathbb{N}}$, such that for every $n \in \mathbb{N}$, $|C_n^1| = |C_n^2|$, $C_n^1 \equiv C_n^2$, and $C_n^1, C_n^2 \in \mathbb{C}_n$, it holds that for any $n = n(\kappa) \leq \mathrm{poly}(\kappa)$:*

$$\mathsf{iO}(C_n^1, 1^\kappa) \overset{T(\kappa)}{\approx} \mathsf{iO}(C_n^2, 1^\kappa).$$

2.4 Input-Hiding Obfuscation

An input-hiding obfuscator for a class of circuits $\mathbb{C}$, as defined by Barak *et al.* [BBC+14], has the security guarantee that given an obfuscation of a randomly drawn circuit in the family $\mathbb{C}$, it is hard for an adversary to find an accepting input. In our work, we consider input-hiding obfuscation for the class of multi-bit point functions. A multi-bit point function $I_{x,y}$ is defined by an input $x \in \{0,1\}^n$, and an output $y \in \{0,1\}^k$. $I_{x,y}$ outputs y on input x, and 0 on all other inputs. Informally, we assume that given the obfuscation of $I_{x,y}$ for a uniformly random x and an arbitrary y, it is hard for an adversary to recover x.

Definition 2.6 (T-secure Input-Hiding Obfuscator [BBC+14])**.** *Let $T : \mathbb{N} \to \mathbb{N}$ be a function, and let $\mathbb{C} = \{\mathbb{C}_n\}_{n\in\mathbb{N}}$ be a family of poly-size circuits, where $\mathbb{C}_n$ is a set of boolean circuits operating on inputs of length n. A* PPT *obfuscator* hideO *is a T-secure input-hiding obfuscator for $\mathbb{C}$, if it satisfies the preserving functionality requirement of Definition 2.5, as well as the following security requirement. For every poly-size (non-uniform) adversary* Adv *and all sufficiently large n,*

$$\Pr_{C \leftarrow \mathbb{C}_n, \mathsf{hideO}} [C(\mathrm{Adv}(\mathsf{hideO}(C))) \neq 0] \leq T^{-1}(n)$$

We emphasize that (unlike other notions of T-security used in this work), we only allow the adversary for a T-secure input hiding obfuscation to run in polynomial time. Nevertheless, depending on the function T, the definition of T-secure input hiding is quite strong. In particular, for the typical case of proof-systems with soundness $2^{n^{\epsilon}}$ (where $\epsilon > 0$ is a constant) we will assume input-hiding obfuscation for $T = 2^{n-n^{\epsilon}}$, which means that a polynomial-time adversary can only do sub-exponentially better than the trivial attack that picks random inputs until it finds an accepting input (this attack succeeds with probability $\text{poly}(n)/2^n$). This is also why we do not separate the security parameter from the input length (the adversary can always succeed with probability 2^{-n}, assuming there exists an accepting input).

We assume input-hiding obfuscation for the class of multi-bit point functions (see above), where the point x is drawn uniformly at random, and the output y is arbitrary. In particular, we do not assume that the collection $\mathbb{C}$ of pairs (x, y) can be sampled efficiently, only that its marginal distribution on x is uniform.

Assumption 2.7 (T-secure Input-Hiding for Multi-Bit Point Functions)**.** *Let $T, k : \mathbb{N} \to \mathbb{N}$ be functions. An obfuscator* hideO *is a T-secure input-hiding obfuscator for (n, k)-multi-bit point functions if for every collection $\mathbb{C}$ as below,* hideO *is a T-secure input-hiding obfuscator for $\mathbb{C}$. In the collection $\mathbb{C}$, for every $n \in \mathbb{N}$, every function $I_{x,y} \in \mathbb{C}_n$ has $x \in \{0,1\}^n, y \in \{0,1\}^{k(n)}$, and the marginal distribution of a random draw from $\mathbb{C}_n$ on x is uniform.*

The assumption is strong in that we do not assume that a random function in $\mathbb{C}$ can be sampled efficiently, or that the output y is an efficient function of the input x. This assumption was studied in [CD08, BC14]. A candidate construction was provided in [CD08]. Loosely speaking, their construction is an extension of the point function obfuscation of Canetti [Can97], where the obfuscation of $I_{x,y}$ consists of a pair of the form (r, r^x), together with k pairs of the form $(r_i, r_i^{\alpha_i})$ where $\alpha_i = x$ if $y_i = 1$ and is uniformly random otherwise. It was proved in [BC14] that this construction is secure in the the generic group model, where the inversion probability is at most $\text{poly}(n) \cdot 2^{-n}$.

2.5 The Fiat-Shamir Paradigm

In this section, we recall the Fiat-Shamir paradigm. For the sake of simplicity of notation, we describe this paradigm when applied to 3-round (as opposed to arbitrary constant round) public-coin protocols. Let $\Pi = (P, V)$ be a 3-round public-coin proof system for an NP language L. We denote its transcripts by (α, β, γ), where β are the messages sent by the verifier, and α, γ are the messages sent by the prover. We denote by n the length of α (i.e., $\alpha \in \{0,1\}^n$), and we denote by k the length of β (i.e., $\beta \in \{0,1\}^k$). We assume that $k \leq \text{poly}(n)$ (since otherwise we can just pad).

Let $\{\mathcal{H}_n\}_{n \in \mathbb{N}}$ be an ensemble of hash functions, such that for every $n \in \mathbb{N}$ and for every $h \in \mathcal{H}_n$,

$$h : \{0,1\}^n \to \{0,1\}^k.$$

We define Π^{FS}, with respect to the hash family $\mathcal{H}$ to be the 2-round protocol obtained by applying the Fiat-Shamir transformation to Π using $\mathcal{H}$. A formal presentation of the "collapsed" protocol $\Pi^{\mathsf{FS}} = (P^{\mathsf{FS}}, V^{\mathsf{FS}})$ is in Figure 2.1.

Protocol $\Pi^{\mathsf{FS}}(1^n, x)$ for an NP Language L

Prover's Input: Statement x and a witness w for $x \in L$.

Verifier's Input: Statement x.

$V^{\mathsf{FS}} \to P^{\mathsf{FS}}$: The verifier V^{FS} chooses a random $h \leftarrow \mathcal{H}_n$, and sends h to the prover P^{FS}.

$P^{\mathsf{FS}} \to V^{\mathsf{FS}}$: The prover P^{FS} simulates an execution with the prover P of Π in the following way:

- Choose a random tape for P and continue the emulation of (P, V) by running P. Let $\alpha \in \{0,1\}^n$ be the first message sent by P in Π.
- Compute $h(\alpha) = \beta$.
- Continue the emulation of P assuming P received β as the second message from V^{FS}. Let γ be the third message sent by P.

Send (α, β, γ) to the verifier V^{FS}.

Verification: The verifier V^{FS} accepts if and only if:

- $h(\alpha) = \beta$.
- V accepts the transcript (α, β, γ).

Figure 2.1: Collapsing a 3-round Protocol $\Pi = (P, V)$ into a 2-round Protocol $\Pi^{\mathsf{FS}} = (P^{\mathsf{FS}}, V^{\mathsf{FS}})$ using $\mathcal{H}$

3 Security of Fiat-Shamir for 3-Message Proofs

We show an instantiation of the Fiat-Shamir paradigm that is sound when it is applied to interactive proofs (as opposed to arguments). Taking n to be a bound on the message lengths of the prover in Π, our instantiation assumes the existence of a 2^n-secure indistinguishability obfuscation scheme iO, a 2^n-secure puncturable PRF family $\mathcal{F}$, and a 2^n-secure input-hiding obfuscation for the class of multi-bit point functions $\mathcal{I}_{n,k}$.

For clarity of exposition, we first show that our instantiation is secure for 3-round public-coin interactive proofs. This is the regime for which the Fiat-Shamir paradigm was originally suggested. We then build on the proof for the 3-message case (or rather the 4-message case, see below), and prove security for any constant number of rounds.

Theorem 3.1 (Fiat-Shamir for 3-message Proofs)**.** *Let Π be a public-coin 3-message interactive proof system, where the lengths of the prover's message are bounded by n, the verifier's message is of length $k \leq \mathrm{poly}(n)$, and the soundness error is negligible.*

Assume the existence of a 2^n-secure puncturable PRF family $\mathcal{F}$, the existence of a 2^n-secure Indistinguishability Obfuscation iO, and the existence of a 2^n-secure input-hiding obfuscation for the class of multi-bit point functions $\{\mathcal{I}_{n,k}\}$. Then the resulting 2-round argument Π^{FS}, obtained by applying the Fiat-Shamir paradigm (see Figure 2.1) to Π with the function family $\mathsf{iO}(\mathcal{F})$, is secure.

In Section 4 we prove the security of the Fiat-Shamir paradigm when applied to any constant round interactive proof. To prove the general (constant round) case, we need to rely on a more general (and more technical) variation of Theorem 3.1. First, we rely on the security of the Fiat-Shamir paradigm for any 4-round interactive proof Π where the first message is sent by the verifier. In the transformed protocol Π^{FS}, the first message of the verifier consists of the first message as in Π, along with a Fiat-Shamir hash function, which will be applied to the prover's first message. In addition, in the generalized theorem we allow the verifier in the original protocol Π to run in time $2^{O(n)}$.

We state the generalized theorem below.

Theorem 3.2 (Theorem 3.1, more General Statement)**.** *Let Π be a 4-message public-coin interactive proof system, where the first message is sent by the verifier, the lengths of the prover's messages are bounded by n, the verifier's messages are of length $k \le \mathrm{poly}(n)$, the soundness error is $\mu(n) = \mathrm{negl}(n)$, and the running time of the verifier is $2^{O(n)}$.*

Let $\epsilon > 0$ be a constant. Assume the existence of a 2^n-secure puncturable PRF family $\mathcal{F}$, the existence of a 2^n-secure Indistinguishability Obfuscation iO, and the existence of a T-secure input-hiding obfuscation for the class of multi-bit point functions $\{\mathcal{I}_{n,k}\}$, where $T = \mu \cdot 2^n \cdot \mathrm{poly}(n)$.

Then the resulting 2-round argument Π^{FS}, obtained by applying the Fiat-Shamir paradigm[10] *to Π with the function family $\mathsf{iO}(\mathcal{F})$, is secure.*

We remark that $\mu \cdot 2^n \cdot \mathrm{poly}(n)$ is a shorthand for a function T such that for every $c > 0$ and all sufficiently large $n \in \mathbb{N}$ it holds that $T(n) \ge \mu(n) \cdot 2^n \cdot n^c$.

Proof of Theorem 3.2. Fix any 4-round interactive proof $\Pi = (P, V)$ as claimed in the theorem statement. Let $\mu = \mathrm{negl}(n)$ be the soundness error of Π. Suppose for the sake of contradiction that there exists a poly-size cheating prover P^* who breaks the soundness of the protocol Π^{FS} with respect to some $x^* \notin L$ with probability $\nu = 1/\mathrm{poly}(n)$.

There must exist a choice for the verifier's first message τ in Π, such that the following two conditions hold: (i) Even conditioned on the first part of the first message in Π^{FS} being τ, the cheating prover P^* still breaks the soundness of the protocol Π^{FS} on x^* with probability at least $(\nu/2)$, and (ii) even conditioned on the first message in Π being τ, the original protocol Π still has soundness error at most $(2\mu/\nu)$. Such a τ must exist because at least a $(\nu/2)$-fraction of the messages must satisfy condition (i) (otherwise P^* cannot break Π^{FS} with total probability ν), and the fraction that do not satisfy condition (ii) must be smaller than $(\nu/2)$ (otherwise the soundness of Π is smaller than μ).

Fix the verifier's first message to always be τ (both in the original and in the transformed protocols). We have that:

$$\Pr_{s,\mathsf{iO}}\left[P^*(\tau, \mathsf{iO}(s)) = (\alpha, \gamma) \text{ s.t. } V(x^*, \tau, \alpha, f_s(\alpha), \gamma) = 1\right] \ge \nu/2, \tag{3.1}$$

where $\mathsf{iO}(s)$ refers to the iO obfuscation of a random function f_s from the family $\mathcal{F}$.

[10] For 4-message proofs, the same paradigm as in Figure 2.1 is used, except that the verifier also sends its first message from the base proof-system (i.e., a random string) in the first round.

The relaxed verifier and its properties. To obtain a contradiction, we analyze a relaxed verifier V' (which is only used in the security analysis). The relaxed verifier accepts a transcript (α, β, γ) if the original verifier V would accept, or if the first $\lceil \log(\nu/(2\mu) \rceil$ bits of β are all 0 (where recall that μ is the soundness error of Π).[11] In particular, whenever V accepts, the relaxed verifier V' also accepts, and so:

$$\Pr_{s,\mathsf{iO}}\Big[P^*(\tau, \mathsf{iO}(s)) = (\alpha, \gamma) \text{ s.t. } V'(x^*, \tau, \alpha, f_s(\alpha), \gamma) = 1\Big] \geq \nu/2. \tag{3.2}$$

We take μ' to be the soundness of the interactive proof (P, V') (after τ is fixed), which runs the relaxed verifier. Observe that by a union bound

$$\mu' \leq (2\mu/\nu) + 2^{-\lceil \log(\nu/(2\mu)) \rceil} \leq 4\mu/\nu,$$

(in particular if μ is negligible, then so is μ').

We define:

$$\mathsf{ACC} \;\; = \;\; \big\{(\alpha, \beta) : \exists \gamma \text{ s.t. } V'(x^*, \tau, \alpha, \beta, \gamma) = 1\big\}$$

Observe that membership in ACC can be computed in time $2^n \cdot \mathrm{poly}(n) = 2^{O(n)}$ by enumerating over all γ's and running V'. Equation (3.2) implies that there exists a poly-size adversary $\mathcal{A}$ (that just outputs the first part of P^*'s output) such that:

$$\Pr_{s,\mathsf{iO}}\Big[\mathcal{A}(\mathsf{iO}(s)) \text{ outputs some } \alpha \text{ s.t. } \big(\alpha, f_s(\alpha)\big) \in \mathsf{ACC}\Big] \geq \nu/2. \tag{3.3}$$

Using Eq. (3.3) we prove our main lemma.

Lemma 3.3.

$$\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\Big[\mathcal{A}\big(\mathsf{iO}(s\{\alpha^*\}, \alpha^*, u^*)\big) = \alpha^* \;\Big|\; \big(\alpha^*, u^*\big) \in \mathsf{ACC}\Big] \geq 2^{-n+2} \cdot \nu/\mu'$$

where α^* and u^* are uniformly distributed (in $\{0,1\}^n$ and $\{0,1\}^k$, respectively) and $\mathsf{iO}(s\{\alpha^*\}, \alpha^*, u^*)$ refers to an iO obfuscation of the program that contains the seed s punctured at the point α^*, and on input α first checks if $\alpha = \alpha^*$ and if so outputs u^* and otherwise outputs $f_s(\alpha)$.

Proof. We prove the lemma by analyzing the probability that the event

$$\Big(\mathcal{A}\big(\mathsf{iO}(s\{\alpha^*\}, \alpha^*, u^*)\big) = \alpha^*\Big) \;\wedge\; \Big(\big(\alpha^*, u^*\big) \in \mathsf{ACC}\Big)$$

occurs.

[11] In the original protocol Π, it may be the case that different messages α sent by the prover can lead the verifier to accept with different probabilities. E.g., some specific α's may lead the verifier to accept with probability μ and others with probability 0. This presents a technical difficulty later in the proof and so we construct the relaxed verifier V' so that every string α leads it to accept with roughly the same probability (up to a small multiplicative constant) without increasing the soundness error by too much.

By the exponential hardness of the puncturable PRF, and the fact that membership in ACC is computable in $2^{O(n)}$ time, we have that

$$\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s\{\alpha^*\},\alpha^*,u^*))=\alpha^*\\ \wedge\\ \big(\alpha^*,u^*\big)\in\mathsf{ACC}\end{array}\right]\geq\Pr_{s,\alpha^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s\{\alpha^*\},\alpha^*,f_s(\alpha^*)))=\alpha^*\\ \wedge\\ \big(\alpha^*,f_s(\alpha^*)\big)\in\mathsf{ACC}\end{array}\right]-2^{-2n}. \tag{3.4}$$

Further applying the exponential hardness of the iO scheme (and the fact that membership in ACC can be decided in $2^{O(n)}$ time), we get that:

$$\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s\{\alpha^*\},\alpha^*,u^*))=\alpha^*\\ \wedge\\ \big(\alpha^*,u^*\big)\in\mathsf{ACC}\end{array}\right]\geq\Pr_{s,\alpha^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s))=\alpha^*\\ \wedge\\ \big(\alpha^*,f_s(\alpha^*)\big)\in\mathsf{ACC}\end{array}\right]-2\cdot2^{-2n}. \tag{3.5}$$

Using elementary probability theory, we have that:

$$\begin{aligned}\Pr_{s,\alpha^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s))=\alpha^*\\ \wedge\\ \big(\alpha^*,f_s(\alpha^*)\big)\in\mathsf{ACC}\end{array}\right]&=\Pr_{s,\alpha^*,\mathsf{iO}}\left[\bigcup_\alpha\Big((\mathcal{A}(\mathsf{iO}(s))=\alpha^*)\wedge\Big(\big(\alpha^*,f_s(\alpha^*)\big)\in\mathsf{ACC}\Big)\wedge(\alpha^*=\alpha)\Big)\right]\\ &=\sum_\alpha\Pr_{s,\alpha^*,\mathsf{iO}}\Big[\Big((\mathcal{A}(\mathsf{iO}(s))=\alpha)\wedge\Big(\big(\alpha,f_s(\alpha)\big)\in\mathsf{ACC}\Big)\wedge(\alpha^*=\alpha)\Big)\Big]\\ &=2^{-n}\sum_\alpha\Pr_{s,\mathsf{iO}}\Big[(\mathcal{A}(\mathsf{iO}(s))=\alpha)\wedge\Big(\big(\alpha,f_s(\alpha)\big)\in\mathsf{ACC}\Big)\Big]\\ &=2^{-n}\Pr_{s,\mathsf{iO}}\Big[\mathcal{A}(\mathsf{iO}(s))\text{ outputs some }\alpha\text{ s.t. }\big(\alpha,f_s(\alpha)\big)\in\mathsf{ACC}\Big]\\ &\geq2^{-n}\cdot\nu/2\end{aligned}$$

where the last inequality is by Eq. (3.3). Thus, we have that:

$$\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s\{\alpha^*\},\alpha^*,u^*))=\alpha^*\\ \wedge\\ \big(\alpha^*,u^*\big)\in\mathsf{ACC}\end{array}\right]\geq\frac{1}{4}\cdot2^{-n}\cdot\nu.$$

By the soundness of the underlying proof-system, it holds that $\Pr_{\alpha^*,u^*}[(\alpha^*,u^*)\in\mathsf{ACC}]\leq\mu'$ (since otherwise a cheating prover could violate soundness by just sending a random α^*).[12] By definition of conditional probability we have that

$$\begin{aligned}\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\Big[\mathcal{A}(\mathsf{iO}(s\{\alpha^*\},\alpha^*,u^*))=\alpha^*\ \Big|\big(\alpha^*,u^*\big)\in\mathsf{ACC}\Big]&=\frac{\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\left[\begin{array}{c}\mathcal{A}(\mathsf{iO}(s\{\alpha^*\},\alpha^*,u^*))=\alpha^*\\ \wedge\\ \big(\alpha^*,u^*\big)\in\mathsf{ACC}\end{array}\right]}{\Pr_{\alpha^*,u^*}[(\alpha^*,u^*)\in\mathsf{ACC}]}\\ &\geq\frac{1}{4}\cdot2^{-n}\cdot\nu/\mu',\end{aligned}$$

and the lemma follows. □

[12] It may at first seem odd that we only use the soundness of the underlying proof-system with respect to a cheating prover that just sends a random message α^*. Recall however that here we consider the *relaxed* verifier who, by design, has a (roughly) similar acceptance probability given any string α.

We are now ready to use (and break) our input-hiding obfuscator hideO. Lemma 3.3, together with the 2^n-security of the iO implies that

$$\Pr_{s,\alpha^*,u^*,\mathsf{iO}}\left[\mathcal{A}(\mathsf{iO}(s,\mathsf{hideO}(\alpha^*,u^*))) = \alpha^* \,\middle|\, (\alpha^*,u^*) \in \mathsf{ACC}\right] \geq \frac{1}{4}\cdot 2^{-n}\cdot \nu/\mu' - 2^{-n} \geq \frac{1}{8}\cdot 2^{-n}\cdot \nu/\mu', \tag{3.6}$$

where α^* and u^* are uniformly distributed and $\mathsf{iO}(s,\mathsf{hideO}(\alpha^*,u^*))$ refers to the iO obfuscation of the program that contains a seed s for a PRF (in its entirety), and the input-hiding obfuscation $\mathsf{hideO}(\alpha^*,u^*)$ of a multi-bit point function that on input α^* outputs u^*. The program uses the input-hiding obfuscation to check if its input equals α^*, and if so outputs the same value as $\mathsf{hideO}(\alpha^*,u^*)$. Otherwise the program behaves like the PRF.

Eq. (3.6) is almost what we want. Namely, an adversary that given access to $\mathsf{hideO}(\alpha^*,u^*)$ produces α^* with probability $\omega(\mathrm{poly}(n)/2^n)$ (since ν is inverse polynomial and μ is a negligible function). The only remaining problem is that the distribution of (α^*,u^*) is not quite what we need. More specifically, in Eq. (3.6) (α^*,u^*) are distributed uniformly conditioned on $(\alpha^*,u^*) \in \mathsf{ACC}$, whereas we need for the marginal distribution of α to be uniform in order to break the hideO obfuscation. Using the properties of the *relaxed* verifier, we show that these two distributions are actually closely related.

We define the following two distributions. The distribution $\mathcal{T}_1$ is obtained by jointly picking a pair (α,β) uniformly from ACC (this is the distribution from which (α^*,u^*) are sampled from in Eq. (3.6)). $\mathcal{T}_2$ is the distribution obtained by picking a uniformly random $\alpha \in \{0,1\}^n$ and then a random β conditioned on $(\alpha,\beta) \in \mathsf{ACC}$ (i.e. the marginal distribution on α is uniform). For $\alpha^* \in \{0,1\}^n$, $\beta^* \in \{0,1\}^k$, we use $\mathcal{T}_1[\alpha^*,\beta^*]$ and $\mathcal{T}_2[\alpha^*,\beta^*]$ to denote the probability of the pair (α^*,β^*) by $\mathcal{T}_1$ and by $\mathcal{T}_2$ (respectively).

Proposition 3.4. *For any $\alpha^* \in \{0,1\}^n$ and $\beta^* \in \{0,1\}^k$:*

$$\mathcal{T}_2[\alpha^*,\beta^*] \geq \frac{1}{4}\mathcal{T}_1[\alpha^*,\beta^*]$$

Proof. For every α^* denote by:

$$S_{\alpha^*} = \left\{\beta^* \in \{0,1\}^k : (\alpha^*,\beta^*) \in \mathsf{ACC}\right\}.$$

By construction of the relaxed verifier V', we know that for every $\alpha \in \{0,1\}^n$ it holds that

$$\frac{\mu}{\nu} \leq \frac{|S_\alpha|}{2^k} \leq \frac{4\mu}{\nu}.$$

In particular, for any $\alpha,\alpha^* \in \{0,1\}^n$:

$$|S_\alpha| \geq \frac{1}{4}|S_{\alpha^*}|.$$

Now we have that:

$$\mathcal{T}_1[\alpha^*,\beta^*] = \frac{1}{\sum_{\alpha\in\{0,1\}^n}|S_\alpha|} \leq \frac{4}{\sum_{\alpha\in\{0,1\}^n}|S_{\alpha^*}|} = \frac{4}{2^n\cdot|S_{\alpha^*}|} = 4\mathcal{T}_2[\alpha^*,\beta^*] \tag{3.7}$$

□

In particular, drawing by $\mathcal{T}_2$ rather than $\mathcal{T}_1$ can only decrease the success probability of $\mathcal{A}$ by a multiplicative factor of 4. Moreover, when drawing by $\mathcal{T}_2$, the marginal distribution on α^* *is* uniform. Thus Proposition 3.4 and Eq. (3.6) imply that there exists a poly-size adversary $\mathcal{A}$, such that

$$\Pr_{(\alpha^*,u^*)\leftarrow\mathcal{T}_2,\mathsf{hideO}}[\mathcal{A}(\mathsf{hideO}(\alpha^*,u^*))=\alpha^*]\geq\frac{1}{32}\cdot\frac{\nu}{\mu'\cdot 2^n}$$

where α^* drawn by $\mathcal{T}_2$ *is* uniformly random. Since ν is an inverse polynomial and $\mu' = O(\mu/\nu)$, this contradicts the $T = \mu \cdot 2^n \cdot \mathrm{poly}(n)$-security of the input-hiding obfuscation hideO.

□

4 Security of Fiat-Shamir for Multi-Round Proofs

In this section we show a secure instantiation of the Fiat-Shamir methodology for transforming any constant-round interactive proof into a 2-round computationally-sound argument. We assume for the sake of simplicity, and without loss of generality, that the verifier always sends the first message, and thus consider interactive protocols with an even number of rounds. Namely, for any constant $c \geq 2$, we consider a $2c$-round interactive proof $\Pi = (P, V)$. We assume without loss of generality that all of the prover's messages are of the same length, and denote this length by n (i.e $\forall i, \alpha_i \in \{0,1\}^n$). Similarly, we assume without loss of generality that all of the verifier's messages are of the same length, and denote this length by k (i.e. $\forall i, \beta_i \in \{0,1\}^k$). We assume without loss of generality that $k \leq n$. All these assumptions are only for the simplicity of notations, and can be easily achieved by padding.

For every $i \in [c-1]$, let $\{\mathcal{F}_n^{(i)}\}_{n\in\mathbb{N}}$ be an ensemble of hash functions, such that for every $n \in \mathbb{N}$ and for every $f^{(i)} \in \mathcal{F}_n$,

$$f : \{0,1\}^{i\cdot(n+k)} \to \{0,1\}^k.$$

We assume without loss of generality that there exists a polynomial p such that for every $i \in [c-1]$ and for every $n \in \mathbb{N}$,

$$\mathcal{F}_n^{(i)} = \{f_s^{(i)}\}_{s\in\{0,1\}^{p(n)}}.$$

We define Π^{FS} to be the 2-round protocol obtained by applying the multi-round Fiat-Shamir transformation to Π using $(\mathsf{iO}(f_{s_1}^{(1)}), \ldots, \mathsf{iO}(f_{s_{c-1}}^{(c-1)}))$, where $f_{s_i}^{(i)} \leftarrow \mathcal{F}_n^{(i)}$ for every $i \in [c-1]$. The security of Π^{FS} is shown in Theorem 4.1 below.

Theorem 4.1 (Fiat-Shamir Transform for Multi-Round Interactive Proofs)**.** *Let $\mu : \mathbb{N} \to [0,1]$ be a function. Assume the existence of a 2^n-secure puncturable PRF family $\mathcal{F}$, assume the existence of a 2^n-secure Indistinguishability Obfuscation, and assume the existence of a $\mu \cdot 2^n \cdot \mathrm{poly}(n)$-secure input-hiding obfuscation for the class of multi-bit point functions $\{\mathcal{I}_{n,k}\}$.*

Then for any constant $c \in \mathbb{N}$ such that $c \geq 2$, and any $2c$-round interactive proof Π with soundness μ, the resulting 2-round argument Π^{FS}, obtained by applying the multi-round Fiat-Shamir transformation to Π with the function family $\mathsf{iO}(\mathcal{F})$, is secure.

Proof. The proof is by induction on $c \in \mathbb{N}$, for $c \geq 2$. The base case $c = 2$ follows immediately from Theorem 3.1. Suppose the theorem statement is true for $< c$ rounds, and we will prove that it is true for c rounds.

To this end, fix any $2c$-round interactive proof Π for proving membership in a language L. Suppose for the sake of contradiction that Π^{FS} is not secure. Namely, there exists a poly-size cheating prover P^* and there exists $x^* \notin L$ such that P^* succeeds in convincing the verifier of Π^{FS} that $x^* \in L$ with non-negligible probability.

Consider the following protocol Ψ for proving membership in L, which consists of $2c-2$ rounds: In the first round the verifier chooses the first message that it would have sent in Π, which we denote by β_0. In addition, it chooses a random seed $s_1 \leftarrow \{0,1\}^{p(n)}$, and sends to the prover the pair $(\beta_0, \mathsf{iO}(f^{(1)}_{s_1}))$. Then, the prover chooses $(\alpha_1, \beta_1, \alpha_2)$ such that $\beta_1 = f^{(1)}_{s_1}(\alpha_1)$, and such that α_1 and α_2 are chosen as in Π. It sends $(\alpha_1, \beta_1, \alpha_2)$ to the verifier. Then the prover and verifier continue to execute the protocol Π interactively, conditioned on $(\beta_0, \alpha_1, \beta_1, \alpha_2)$. Finally, the verifier accepts if and only if the verifier of Π would have accepted the resulting transcript and $\beta_1 = f^{(1)}_{s_1}(\alpha_1)$.

If Ψ is a sound proof, then by our induction hypothesis $(\Psi)^{\mathsf{FS}}$ is sound. However, note that P^* can be trivially converted into a cheating prover that breaks the soundness of $(\Psi)^{\mathsf{FS}}$, contradicting our induction hypothesis that the Fiat-Shamir transformation is sound for interactive proofs with $2(c-1)$ rounds (with the function family $\mathsf{iO}(\mathcal{F})$). Thus, it must be the case that Ψ is not a sound proof. Namely, there exists a poly-size cheating prover P^{**}, an element $x^* \notin L$, and a polynomial q, such that P^{**} convinces the verifier of Ψ to accept x^* with probability $\geq 1/q(\kappa)$ for infinitely many $\kappa \in \mathbb{N}$.

Consider the 4-round protocol Φ, which consists of the first 4 rounds of Π, denoted by $(\beta_0, \alpha_1, \beta_1, \alpha_2)$. Given a transcript $(\beta_0, \alpha_1, \beta_1, \alpha_2)$ the verifier of Φ accepts if and only if there exists a strategy of the (cheating) prover of Π that causes the verifier of Π to accept with probability $\geq 1/q(\kappa)$ conditioned on the first 4-rounds of Π being $(\beta_0, \alpha_1, \beta_1, \alpha_2)$. Note that the verifier of Φ runs in time $\text{poly}(2^{c(n+k)}) = 2^{O(n)}$. The soundness of Π implies that Φ is also sound. Note however that Φ^{FS} is not sound since P^{**} can be used to break the soundness of Φ^{FS}. This is in contradiction to Theorem 3.2.

□

Acknowledgments

We thank an anonymous reviewer for suggesting, and allowing us to use, a significant simplification to our original proof.

This work was done in part while the authors were visiting the Simons Institute for the Theory of Computing, supported by the Simons Foundation and by the DIMACS/Simons Collaboration in Cryptography through NSF grant #CNS-1523467.

References

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

[BBC+14] Boaz Barak, Nir Bitansky, Ran Canetti, Yael Tauman Kalai, Omer Paneth, and Amit Sahai. Obfuscation for evasive functions. In *TCC*, pages 26–51, 2014.

[BC14] Nir Bitansky and Ran Canetti. On strong simulation and composable point obfuscation. *J. Cryptology*, 27(2):317–357, 2014.

[BCC+14] Nir Bitansky, Ran Canetti, Henry Cohn, Shafi Goldwasser, Yael Tauman Kalai, Omer Paneth, and Alon Rosen. The impossibility of obfuscation with auxiliary input or a universal simulator. In *CRYPTO*, pages 71–89, 2014.

[BDG+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 *TCC*, pages 182–201, 2013.

[BDNP08] Assaf Ben-David, Noam Nisan, and Benny Pinkas. Fairplaymp: a system for secure multi-party computation. In *ACM Conference on Computer and Communications Security*, pages 257–266, 2008.

[BGI+12] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. *J. ACM*, 59(2):6, 2012.

[BGI14] Elette Boyle, Shafi Goldwasser, and Ioana Ivan. Functional signatures and pseudorandom functions. In *PKC*, pages 501–519, 2014.

[Blu87] Manuel Blum. How to prove a theorem so no one else can claim it. In *Proceedings of the International Congress of Mathematicians*, pages 1444–1451, 1987.

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

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *ACM Conference on Computer and Communications Security*, pages 62–73, 1993.

[BW13] Dan Boneh and Brent Waters. Constrained pseudorandom functions and their applications. In *ASIACRYPT*, pages 280–300, 2013.

[Can97] Ran Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*, pages 455–469, 1997.

[CCR15] Ran Canetti, Yilei Chen, and Leonid Reyzin. On the correlation intractability of obfuscated pseudorandom functions. *IACR Cryptology ePrint Archive*, 2015:334, 2015.

[CD08] Ran Canetti and Ronny Ramzi Dakdouk. Obfuscating point functions with multibit output. In *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Istanbul, Turkey, April 13-17, 2008. Proceedings*, pages 489–508, 2008.

[DNRS99] Cynthia Dwork, Moni Naor, Omer Reingold, and Larry J. Stockmeyer. Magic functions. In *FOCS*, pages 523–534, 1999.

[DRV12] Yevgeniy Dodis, Thomas Ristenpart, and Salil P. Vadhan. Randomness condensers for efficiently samplable, seed-dependent sources. In *TCC*, pages 618–635, 2012.

[FS86] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In *CRYPTO*, pages 186–194, 1986.

[GGH+13] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*, pages 40–49, 2013.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, 1986.

[GK03] Shafi Goldwasser and Yael Tauman Kalai. On the (in)security of the fiat-shamir paradigm. In *FOCS*, pages 102–113, 2003.

[GK05] Shafi Goldwasser and Yael Tauman Kalai. On the impossibility of obfuscation with auxiliary input. In *FOCS*, pages 553–562, 2005.

[GK16] Shafi Goldwasser and Yael Tauman Kalai. Cryptographic assumptions: A position paper. In *Theory of Cryptography - 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part I*, pages 505–522, 2016.

[GLSW14] Craig Gentry, Allison B. Lewko, Amit Sahai, and Brent Waters. Indistinguishability obfuscation from the multilinear subgroup elimination assumption. *IACR Cryptology ePrint Archive*, 2014:309, 2014.

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

[HT98] Satoshi Hada and Toshiaki Tanaka. On the existence of 3-round zero-knowledge protocols. In *CRYPTO*, pages 408–423, 1998.

[KPTZ13] Aggelos Kiayias, Stavros Papadopoulos, Nikos Triandopoulos, and Thomas Zacharias. Delegatable pseudorandom functions and applications. In *ACM CCS*, pages 669–684, 2013.

[Mic94] Silvio Micali. CS proofs. In *FOCS*, pages 436–453, 1994.

[MNPS04] Dahlia Malkhi, Noam Nisan, Benny Pinkas, and Yaron Sella. Fairplay - secure two-party computation system. In *USENIX Security Symposium*, pages 287–302, 2004.

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

[PS96] David Pointcheval and Jacques Stern. Security proofs for signature schemes. In *EUROCRYPT*, pages 387–398, 1996.

[RR97] Alexander A. Razborov and Steven Rudich. Natural proofs. *J. Comput. Syst. Sci.*, 55(1):24–35, 1997.

[SW14] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: deniable encryption, and more. In *STOC*, pages 475–484, 2014.