Theoretical Computer Science 740 (2018) 28–62

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Fiat–Shamir for highly sound protocols is instantiable [☆]

Arno Mittelbach [a], Daniele Venturi [b,*]

[a] *Cryptoplexity, Technische Universität Darmstadt, Germany*
[b] *Department of Computer Science, Sapienza University of Rome, Italy*

ARTICLE INFO

Article history:
Received 6 July 2016
Received in revised form 25 April 2018
Accepted 1 May 2018
Communicated by G. Persiano

Keywords:
Fiat–Shamir transform
Non-interactive zero knowledge
Signature schemes
Indistinguishability obfuscation
Standard model

ABSTRACT

The Fiat–Shamir (FS) transformation (Fiat and Shamir, Crypto '86) is a popular paradigm for constructing very efficient non-interactive zero-knowledge (NIZK) arguments and signature schemes from a hash function and any three-move interactive protocol satisfying certain properties. Despite its wide-spread applicability both in theory and in practice, the known positive results for proving security of the FS paradigm are in the random oracle model only, i.e., they assume that the hash function is modeled as an external random function accessible to all parties. On the other hand, a sequence of negative results shows that for certain classes of interactive protocols, the FS transform cannot be instantiated in the standard model.

We initiate the study of complementary positive results, namely, studying classes of interactive protocols where the FS transform *does* have standard-model instantiations. In particular, we show that for a class of "highly sound" protocols that we define, instantiating the FS transform via a q-wise independent hash function yields NIZK arguments and secure signature schemes. In the case of NIZK, we obtain a weaker "q-bounded" zero-knowledge flavor where the simulator works for all adversaries asking an a-priori bounded number of queries q; in the case of signatures, we obtain the weaker notion of random-message unforgeability against q-bounded random message attacks.

Our main idea is that when the protocol is highly sound, then instead of using random-oracle programming, one can use complexity leveraging. The question is whether such highly sound protocols exist and if so, which protocols lie in this class. We answer this question in the affirmative in the common reference string (CRS) model and under strong assumptions. Namely, assuming indistinguishability obfuscation and puncturable pseudorandom functions we construct a compiler that transforms any 3-move interactive protocol with instance-independent commitments and simulators (a property satisfied by the Lapidot–Shamir protocol, Crypto '90) into a compiled protocol in the CRS model that is highly sound. We also present a second compiler, in order to be able to start from a larger class of protocols, which only requires instance-independent commitments (a property for example satisfied by the classical protocol for quadratic residuosity due to Blum, Crypto '81). For the second compiler we require dual-mode commitments.

We hope that our work inspires more research on classes of (efficient) 3-move protocols where Fiat–Shamir is (efficiently) instantiable.

[☆] An abridged version of this paper appears in the Proceedings of the 10th Conference on Security and Cryptography for Networks—SCN 2016. This is the full version.

[*] Corresponding author.
E-mail addresses: mail@arno-mittelbach.de (A. Mittelbach), venturi@di.uniroma1.it (D. Venturi).

https://doi.org/10.1016/j.tcs.2018.05.001
0304-3975/© 2018 Elsevier B.V. All rights reserved.

1. Introduction

The Fiat–Shamir (FS) transformation [37] is a popular[1] technique to build efficient non-interactive zero-knowledge (NIZK) arguments and signature schemes, starting from three-round *p*ublic-coin (3PC) protocols satisfying certain properties. In a 3PC protocol the prover (with statement x and witness w) starts by sending a commitment α, to which the verifier replies with a challenge β drawn at random from some space $\mathcal{B}$; finally the prover sends a reply γ and the verifier's verdict is computed as a predicate of the transcript (α, β, γ) and the statement x being proven.[2]

1.1. Fiat–Shamir NIZK and signatures

We briefly review the two main applications of the FS transform below.

NIZK A NIZK is a non-interactive protocol in which the prover—holding a witness w for membership of a statement x in some *NP*-language L—can convince the verifier—holding just x—that $x \in L$, by sending a single message π. NIZK should satisfy three properties. First, *completeness* says that an honest prover holding a valid witness (almost) always convinces an honest verifier. Second, *soundness* says that a malicious prover should not be able to convince the honest verifier into accepting a *false* statement, i.e. a statement $x \notin L$; we speak of*arguments* (resp., *proofs*) when the soundness requirement holds for all computationally bounded (resp., computationally unbounded) provers. Third, *zero knowledge* requires that a proof does not reveal anything about the witness beyond the validity of the statement being proven.

NIZK require a setup assumption, typically in the form of a common reference string (CRS). Starting from a 3PC protocol, the FS transform makes it a NIZK by having the prover compute the verifier's challenge as a hash of the commitment α via some hash function H (with "hash key" hk); this results in a single message $\pi = (\alpha, \beta, \gamma)$, where $\beta = \mathsf{H}(\mathsf{hk}, \alpha)$, that is sent from the prover to the verifier.[3] (The description of the hash function, i.e. key hk, is included as part of the CRS.)

Apart from being a fascinating topic, NIZK have been demonstrated to be extremely useful for cryptographic applications (see, e.g., [39,49,33,19,18,28]).

Signatures Digital signatures are among the most important and well-studied cryptographic tools. Signature schemes allow a signer (holding a public/secret key pair $(\mathsf{pk}, \mathsf{sk})$) to generate a signature σ on a message m, in such a way that anyone possessing the public key pk can verify the validity of (m, σ). Signatures must be unforgeable, meaning that it should be hard to forge a signature on a "fresh" chosen message (even after seeing polynomially many signatures on possibly chosen messages).

Starting with a 3PC protocol, the FS transform makes it a signature scheme by having the signer compute the verifier's challenge as a hash of the commitment α, concatenated with the message m, via some hash function H (with "hash key" hk); this results in a signature $\sigma = (\alpha, \beta, \gamma)$, where $\beta = \mathsf{H}(\mathsf{hk}, \alpha || m)$.

1.2. Positive and negative results

We refer to the non-interactive system obtained by applying the FS transform to a 3PC protocol (i.e., a NIZK or a signature scheme) as the *FS collapse*. A fundamental question in cryptography is to understand what properties the initial 3PC protocol and the hash function should satisfy in order for the FS collapse to be a NIZK argument or a secure signature scheme. This question has been studied extensively in the literature; we briefly review the current state of affairs below.

Positive results All security proofs for the FS transform follow the random oracle methodology (ROM) of Bellare and Rogaway [12], i.e., they assume that the function H behaves like an external random function accessible to all parties (including the adversary). In particular, a series of papers [37,53,55,1] establishes that the FS transform yields a secure signature scheme in the ROM provided that the starting 3PC is a passively secure identification scheme. The first definition of NIZK in the ROM dates back to [12] (where a particular protocol was analyzed); in general, it is well known that, always in the ROM, the FS transform yields a NIZK satisfying sophisticated properties such as simulation-soundness [36] and simulation-extractability [11].

Barak et al. [10] put forward a new hash function property (called entropy preservation[4]) that allows to prove soundness of the FS transformation without random oracles; their result requires that the starting 3PC protocol is statistically sound, i.e. it is a *proof*. Dodis et al. [31] show that such hash functions exist in case a conjecture on the existence of certain "condensers for leaky sources" turns out to be true. Canetti et al. [16] study the correlation intractability of obfuscated pseudorandom functions and show a close connection between entropy preservation and correlation intractability, but it remains open

[1] There are over 3.000 Google-Scholar-known citations to [37], as we type.

[2] Protocols with this shape are sometimes known as Sigma protocols; however, the definition of Sigma protocols typically assumes that the underlying protocol satisfies certain security property which will be slightly different from the ones we need.

[3] The value β is typically omitted from the proof, as the verifier can compute it by itself.

[4] Entropy preservation roughly says that for all efficient adversaries that get a uniformly random hash key hk and produce a correlated value α, the conditional Shannon entropy of $\beta = \mathsf{H}(\mathsf{hk}, \alpha)$ given α, but not hk, is sufficiently large.