Impossibility on the Schnorr Signature from the One-more DL Assumption in the Non-programmable Random Oracle Model*

Masayuki Fukumitsu[1] and Shingo Hasegawa[2]

[1] Faculty of Information Media, Hokkaido Information University,
Nishi-Nopporo 59-2 Ebetsu, Hokkaido, 069–8585 Japan.
fukumitsu@do-johodai.ac.jp
[2] Graduate School of Information Sciences, Tohoku University,
41 Kawauchi, Aoba-ku, Sendai, Miyagi, 980–8576 Japan.
shingo.hasegawa.b7@tohoku.ac.jp

Abstract. The Schnorr signature is one of the representative signature schemes and its security was widely discussed. In the random oracle model (ROM), it is provable from the DL assumption, whereas there is a negative circumstantial evidence in the standard model. Fleischhacker, Jager and Schröder showed that the tight security of the Schnorr signature is unprovable from a strong cryptographic assumption, such as the One-More DL (OM-DL) assumption and the computational and decisional Diffie-Hellman assumption, in the ROM via a generic reduction as long as the underlying cryptographic assumption holds. However, it remains open whether or not the impossibility of the provable security of the Schnorr signature from a strong assumption via a *non-tight* and reasonable reduction. In this paper, we show that the security of the Schnorr signature is unprovable from the OM-DL assumption in the non-programmable ROM as long as the OM-DL assumption holds. Our impossibility result is proven via a non-tight Turing reduction.

Keywords: Schnorr signature · Non-programmable random oracle model · Impossibility result · One-more DL assumption

1 Introduction

The Schnorr signature is one of the representative signature schemes and its security was discussed in several literatures. Pointcheval and Stern [PS00] showed that it is provable to be strongly existentially unforgeable against the chosen message attack (seuf-cma) in the random oracle model (ROM) from the discrete logarithm (DL) assumption. Abdalla, An, Bellare and Namprempre [AABN08] expands their result to cover other signatures which can be obtained via the Fiat-Shamir transformation [FS87] as well as the Schnorr signature.

On the other hand, there is a negative circumstantial evidence for the provable security of the Schnorr signature in the *standard model*. The Schnorr signature is unprovable to be secure from the DL assumption in the standard model via an algebraic reduction as long as the One-More DL (OM-DL) assumption holds [PV05]. The OM-DL assumption [BNPS03] is parameterized by a polynomial T. It intuitively states that any probabilistic polynomial-time (PPT) adversary $\mathcal{A}$ cannot find the DLs $(x_1, x_2, \ldots, x_{T+1})$ of given $T + 1$ group elements $(y_1, y_2, \ldots, y_{T+1})$, even when $\mathcal{A}$ adaptively obtains at most T DLs of arbitrary elements. We occasionally call such a OM-DL assumption *T-OM-DL assumption* explicitly.

For the provable security of the Schnorr signature, the affirmative results were given in the ROM, whereas the impossibility result was given in the standard model. The ROM is different from the standard model in the feature that the hash value is a truly random string in the ROM. This feature enables a reduction in a security proof to simulate the random oracle which involves the *programming technique* [PS00]. The programming technique allows a reduction $\mathcal{R}$, which is constructed in the security proof, to simulate the random oracle by setting hash values itself. By employing this technique, many cryptographic schemes e.g. [Cor02,KK12] were proven to be secure in the ROM. Especially, the forking lemma [PS00] can be realized by using this technique to construct security proofs of several cryptographic schemes including the Schnorr signature.

On the theoretical cryptography, one of the interests is how one can relax the property of the ROM on proving the security of cryptographic schemes. For this purpose, intermediate security models between the ROM and the standard model were proposed. One of these is a *Non-programmable ROM (NPROM)*. The concept of the NPROM was introduced by Nielsen [Nie02] to give the impossibility result on a non-interactive non-committing encryption. They defined the notion in the simulation-based security model. Fischlin, Lehmann, Ristenpart, Shrimpton, Stam and Tessaro [FLR+10] formalized the NPROM for the game-based security proof,

* The preliminary version of this paper appeared in [FH17].

Table 1. The affirmative results on proving the security of the Schnorr signature.

	Model	Security	Tight	Assumption	Forger
[AABN08,PS00]	ROM	seuf-cma		DL	
[FPS20]	ROM	euf-cma	√	DL	AGM
[PV05]	Standard	ukb-cma	√	OM-DL	

and discussed the security of a trapdoor-permutation-based key-encapsulation and a full-domain hash in the NPROM. Most recently, Fischlin, Harasser and Janson [FHJ20] evaluated the security of several OR proofs in this model. In the NPROM, the random oracle is dealt with the independent party, and any parties in the security proof such as a reduction $\mathcal{R}$ and an adversary obtain hash values from the external random oracle as well as the ROM. However, $\mathcal{R}$ is prohibited to simulate it, namely $\mathcal{R}$ cannot set the hash values and hence we cannot use the programming technique.

The security of the Schnorr signature in the NPROM was also discussed. Fischlin and Fleischhacker [FF13] first gave a negative circumstantial evidence. They showed that the Schnorr signature is unprovable to be euf-cma from the DL assumption in the NPROM via a single-instance (SI) reduction as long as the OM-DL assumption holds. Subsequently, their impossibility result was extended to cover other signatures or assumptions [FH16,FH18,ZCC+15]. In particular, Fukumitsu and Hasegawa [FH16] proved that the DL assumption is incompatible with the euf-cma security of the Schnorr signature in the NPROM via a sequentially multi-instance (SMI) reduction. In other words, the Schnorr signature may be unprovable to be euf-cma from the DL assumption in the NPROM as long as the DL assumption holds. The SMI reduction is a reduction such that it can invoke an adversary of the target cryptographic scheme polynomially many times, although it is prohibited to invoke the clones of the adversary concurrently.

As described above, it seems to be hard for the Schnorr signature to prove the security under the DL assumption without the programming technique. One can consider the possibility of security proofs with a cryptographic assumption which is stronger than the DL assumption, such as the OM-DL assumption and the computational and decisional Diffie-Hellman assumption [Bon98]. This question was also discussed and the affirmative results are collected in Table 1. Paillier and Vergnaud [PV05] showed that the Schnorr signature is provable to be unkeybreakable against the chosen-message attack (ukb-cma) from the OM-DL assumption in the standard model. Although it was proven via a tight reduction, the ukb-cma is a weaker notion than the ordinary euf-cma. Fuchsbauer, Plouviez and Seurin [FPS20] prove the euf-cma security of the Schnorr signature in the ROM with the tight security by restricting a computational model of a forger to the algebraic group model (AGM).

Not only the affirmative result, but also the negative circumstantial evidences were given in several literatures. Fleischhacker, Jager and Schröder [FJS14] showed that the Schnorr signature is unprovable to be universally unforgeable against the key only attack (uuf-koa) from some cryptographic assumption, such as not only the DL assumption, but also the OM-DL assumption, in the ROM via a tight and generic reduction as long as the underlying cryptographic assumption holds. Recall that Paillier and Vergnaud [PV05] also gave the impossibility result from the DL assumption via a tight and algebraic reduction. It should be noted that their impossibility results do not contradict to the affirmative results by [AABN08,PS00]. This is because the results by [AABN08,PS00] considered a non-tight reduction for the security proofs, whereas these impossibility results hold only for *tight* and some *constrained* reductions like generic or algebraic reductions. For the impossibility result which concerns a non-tight reduction, Fukumitsu and Hasegawa [FH20] showed that the generalized OM assumption [ZZC+14], which is a generalization of the OM-DL assumption, seems not to imply the euf-cma security of several Fiat-Shamir-type signature schemes. Non-tight reductions are considered in their result, however, they only considered a vanilla reduction which only can invoke a forger once and is prohibited to rewind it. Eventually, it remains open whether or not the impossibility holds on the provable security of the Schnorr signature in the NPROM via a *non-restricted* reduction. The impossibility results mentioned above are collected in Table 2. Note that "only" in the Tight column means that the corresponding impossibility result excludes a tight reduction only.

1.1 Our Contributions

In this paper, we aim to extend the impossibility results concerning the security of the Schnorr signature via a non-restricted reduction. We give an impossibility result on the provable security of the Schnorr signature from the OM-DL assumption in the NPROM via a Turing reduction. It is given by the following theorem.

Table 2. The impossibility results on proving the security of the Schnorr signature

	Model	Assumed reduction $\mathcal{R}$				Resulting meta-reduction $\mathcal{M}$
		Security	Tight	Assumption	Type	Assumption
[PV05]	ROM	uuf-koa	only	DL	algebraic	OM-DL
[GBL08]	ROM	uuf-koa	only	DL	algebraic	OM-DL
[Seu12]	ROM	uuf-koa	only	DL	algebraic	OM-DL
[FJS14]	ROM	uuf-koa	only	DL	generic	DL
[FJS14]	ROM	uuf-koa	only	OM-DL	generic	OM-DL
[ours]	**NPROM-PROM**	**uuf-koa**		**OM-DL**	**Turing**	**OM-DL**
[FF13]	NPROM-NPROM	euf-cma		DL	SI	OM-DL
[FH16]	NPROM-NPROM	suf-sma [BJLS16]		DL	SMI	DL
[FH20]	NPROM-NPROM	euf-cma		generalized OM	Vanilla	generalized OM
[PV05]	Standard	uuf-koa		DL	algebraic	OM-DL

Theorem 4. (Informal) *The Schnorr signature is unprovable to be uuf-koa from the OM-DL assumption in the NPROM via a Turing reduction as long as the OM-DL assumption holds. The considered situations are explained below.*

Theorem 4 is proven intuitively as follows. Assume that there exists a PPT Turing reduction algorithm $\mathcal{R}$ which solves the OM-DL problem by invoking a uuf-koa forger $\mathcal{F}$ of the Schnorr signature in the NPROM. We shall construct a PPT *meta-reduction* algorithm [BV98] $\mathcal{M}$ which solves the OM-DL problem by running $\mathcal{R}$. This means that the OM-DL assumption is broken if there exists such a reduction $\mathcal{R}$. The theorem follows from the contraposition. $\mathcal{M}$ aims to make the assumed reduction $\mathcal{R}$ to solve the OM-DL problem by simulating $\mathcal{F}$.

There are three matters to be considered in the proof of Theorem 4. The first one is that we focus on the *uuf-koa* security [GMR88]. The uuf-koa security informally states that any PPT forger $\mathcal{F}$ cannot find a signature σ of m on a given pair (pk, m) of a public key pk and a message m. In the uuf-koa security, a forger $\mathcal{F}$ makes no signing oracle query. Since uuf-koa is weaker than euf-cma [GMR88], putting together with Theorem 4, the impossibility of the ordinary euf-cma security also follows.

The second one is the hypothetical uuf-koa forger $\widetilde{\mathcal{F}}$ which is provided to $\mathcal{R}$. $\mathcal{R}$ is assumed to solve the OM-DL problem no matter what the description of $\widetilde{\mathcal{F}}$ is. We consider the specific uuf-koa forger $\widetilde{\mathcal{F}}$ which is deterministic and makes only one query to the external random oracle. Since $\widetilde{\mathcal{F}}$ performs the key only attack, $\widetilde{\mathcal{F}}$ makes no query to the signing oracle. And the hash value cannot be controlled by $\mathcal{R}$ because we consider the NPROM setting. Therefore the behavior of $\widetilde{\mathcal{F}}$ is determined totally by the input (pk, m) which is given from $\mathcal{R}$ when $\mathcal{R}$ invokes $\widetilde{\mathcal{F}}$, and the output σ of $\widetilde{\mathcal{F}}$ is also fixed. Thus $\mathcal{R}$ cannot affect $\widetilde{\mathcal{F}}$ even if $\mathcal{R}$ rewinds $\mathcal{F}$ or invokes it concurrently, and then the meta-reduction $\mathcal{M}$ can simulate the hypothetical uuf-koa forger $\widetilde{\mathcal{F}}$ against the Turing reduction $\mathcal{R}$. The way of simulating $\widetilde{\mathcal{F}}$ in $\mathcal{M}$ is described below.

The third one is the treatment of the random oracle. Recall that the reduction $\mathcal{R}$ in the NPROM is modeled to obtain any hash value from the external random oracle to inhibit the programming by $\mathcal{R}$. According to the use of the random oracle by a meta-reduction $\mathcal{M}$, the construction of conventional meta-reductions is divided into the following two types. The former is that $\mathcal{M}$ also obtains any hash value from the external random oracle model. The impossibility results by [FF13,FH16,FH18,FH20] are categorized in this type. We refer to this type of meta-reductions as the *NPROM-NPROM* model. The latter is that $\mathcal{M}$ can simulate the random oracle. This type is referred to as the *NPROM-PROM* model. In fact, Fischlin, Harasser, and Janson [FHJ20] succeeded in the construction of $\mathcal{M}$ by considering the NPROM-PROM model implicitly. In Theorem 4, we employ the NPRO-PROM model to construct our meta-reduction. It remains open whether or not the impossibility result holds in the NPROM-NPROM model via a non-restricted Turing reduction.

In the construction of $\mathcal{M}$, $\mathcal{M}$ aims to make the assumed reduction $\mathcal{R}$ to solve the OM-DL problem. Since $\mathcal{R}$ may invoke a uuf-koa forger $\mathcal{F}$ with a pair (pk, m), $\mathcal{M}$ is required to simulate it. Namely, $\mathcal{M}$ needs to return a valid signature σ of m under pk. In order to simulate $\mathcal{F}$, we utilize the honest-verifier zero-knowledge property of Schnorr signature and the feature of the NPROM-PROM model. The honest-verifier zero-knowledge property derives from the underlying ID scheme of Schnorr signature and this property states that the distribution of transcripts by the underlying ID scheme can be simulated without the secret key. By cooperating this property with the programming of the random oracle by $\mathcal{M}$, we can succeed in simulating the uuf-koa forger in the meta-reduction.

1.2 Related Works

Pass [Pas11] gave the impossibility result of the provable security on the Schnorr ID scheme [Sch91] from which the Schnorr signature is derived via the Fiat-Shamir transformation. They showed that the Schnorr ID is unprovable to be secure against the impersonation under the active attack (imp-aa secure) from several interactive assumptions such as the OM-DL assumption. Note that the imp-aa security of the Schnorr ID was proven from the OM-DL assumption in [BP02]. The difference between these two results is due to the parameter T of the OM-DL assumption. Bellare and Palacio [BP02] considered the case where T is equivalent to the number of the access to the oracle in the imp-aa game, whereas Pass considered that T is asymptotically smaller than the number of the oracle access. It is known that the T_1-OM-DL assumption may be strictly weaker than the T_2-OM-DL assumption when $T_2 > T_1$ [BMV08]. These imply that the Schnorr ID may be unprovable to be secure from the T-OM-DL assumption where the parameter T is strictly smaller than the number of the oracle access.

Although they focused on the provable security of the Schnorr ID, their result seems not to directly elucidate the question on the provable security of the Schnorr signature from the OM-DL assumption in the NPROM. This is because the relationship between the security of the Schnorr signature in the NPROM and the security of the Schnorr ID has not been known so far. Therefore, we consider this question by directly observing the relationship between the security of the Schnorr signature and the OM-DL assumption.

1.3 Differences from Proceedings Version

The earlier version of this paper appeared in [FH17]. In the proceeding version, we showed the impossibility result concerning the selectively unforgeability against the chosen message attack (suf-cma), whereas we show in this paper the impossibility on the uuf-koa security. The difference is due to the new proof technique by [FHJ20]. By employing the new proof technique and the NPROM-PROM model, we can strengthen the result from the impossibility on the suf-cma security to the one on the uuf-koa security.

2 Preliminaries

For any natural number n, let $\mathbb{Z}_n$ denote the residue ring $\mathbb{Z}/n\mathbb{Z}$. The notation $x \in_{\mathrm{U}} X$ means that an element x is sampled uniformly at random from the finite set X. For a finite set X, let $U(X)$ be the uniform distribution over X. And, $x \in_D X$ means that x is sampled according to the distribution D. We denote by $x := y$ that x is defined or substituted as y. For any algorithm $\mathcal{A}$, we define by $y \leftarrow \mathcal{A}(x)$ that $\mathcal{A}$ takes x as input and then outputs y. When $\mathcal{A}$ is probabilistic, we write $y \leftarrow \mathcal{A}(x; r)$ to denote that $\mathcal{A}$ takes x as input with a randomness r and then outputs y, and $\mathcal{A}(x)$ is the random variable on the fixed input x, where the probability is taken over the internal coin flips of $\mathcal{A}$. A function ϵ is *negligible* if for any polynomial ν, there exists a natural number λ_0 such that for any $\lambda > \lambda_0$, $\epsilon(\lambda) < 1/\nu(\lambda)$. For any ensembles $\left\{D_\lambda^{(1)}\right\}_\lambda$ and $\left\{D_\lambda^{(2)}\right\}_\lambda$ of distributions over an ensemble $\{X_\lambda\}_\lambda$ of sets, we say that $\left\{D_\lambda^{(1)}\right\}_\lambda$ is *statistically close* to $\left\{D_\lambda^{(2)}\right\}_\lambda$ if the statistical distance between $\left\{D_\lambda^{(1)}\right\}_\lambda$ and $\left\{D_\lambda^{(2)}\right\}_\lambda$ is negligible in λ, where the statistical distance between $\left\{D_\lambda^{(1)}\right\}_\lambda$ and $\left\{D_\lambda^{(2)}\right\}_\lambda$ is defined as $1/2 \sum_{x \in X_\lambda} \left| \Pr\left[x \in_{D_\lambda^{(1)}} X_\lambda\right] - \Pr\left[x \in_{D_\lambda^{(2)}} X_\lambda\right] \right|$.

Let L denote a key-value list. For any key string $x \in \{0,1\}^*$, $\mathsf{L}[x]$ stands for the value of x. For a string x, $\mathsf{L}[x] = \perp$ means that the value of x is not defined. For any list L, any algorithm or distribution $\mathcal{D}$ and any string x, we denote by $y \leftsquigarrow_{\mathcal{D}} \mathsf{L}[x]$ the lazy sampling from $\mathcal{D}$, in a sense that $y := \mathsf{L}[x]$ if $\mathsf{L}[x] \neq \perp$, or $y := \mathsf{L}[x] \leftarrow \mathcal{D}(x)$ otherwise.

2.1 Signature Scheme

A signature scheme Sig consists of a tuple $(\mathsf{KGen}, \mathsf{Sign}, \mathsf{Ver})$ of three polynomial-time algorithms. KGen is a probabilistic polynomial-time (PPT) key generator which takes a security parameter 1^λ as input, and then outputs a pair (sk, pk) of a secret key and a public key. Sign is a PPT signing algorithm which takes a key pair (sk, pk) and a message m as input, and then outputs a signature σ. Ver is a deterministic verification algorithm which takes a public key pk, a message m and a signature σ as input, and then outputs 1 if σ is a *valid* signature on the message m under the public key pk.

We now introduce for $\mathsf{Sig} := (\mathsf{KGen}, \mathsf{Sign}, \mathsf{Ver})$, the notions of the existential unforgeability against the chosen message attack (euf-cma) and the universal unforgeability against the key-only attack (uuf-koa). Let Q_s be a polynomial in a security parameter λ. The *Q_s-euf-cma game* is defined in the following way: on a security parameter λ,

EF Init A forger $\mathcal{F}$ is given a public key pk where a challenger C generates $(sk, pk) \leftarrow \mathsf{KGen}(1^\lambda)$.
Signing Oracle When $\mathcal{F}$ hands an i-th message $\overline{m}_i$ to C, C replies its signature $\overline{\sigma}_i \leftarrow \mathsf{Sign}(sk, pk, \overline{m}_i)$. Note that $\mathcal{F}$ can access this phase at most Q_s times.
EF Challenge When $\mathcal{F}$ finally returns a pair (m^*, σ^*), C outputs 1 if $m^* \notin \{\overline{m}_i\}_{i=1}^{Q_s}$ and $\mathsf{Ver}(pk, m^*, \sigma^*) = 1$.

In a similar manner, the *uuf-koa game* is defined in the following way: on a security parameter λ,

UF Init A forger $\mathcal{F}$ is given a public key pk and a message m where a challenger C generates $(sk, pk) \leftarrow \mathsf{KGen}(1^\lambda)$ and samples m.
UF Challenge When $\mathcal{F}$ finally returns a signature σ, C outputs 1 if $\mathsf{Ver}(pk, m, \sigma) = 1$.

Let sec $\in \{Q_s$-euf-cma, uuf-koa$\}$. Then $\mathcal{F}$ is said to *win the sec game of* Sig if C outputs 1 in the corresponding game. The signature scheme Sig is said to be *sec* if any PPT forger $\mathcal{F}$ wins the corresponding game with a negligible probability. The probability is taken over the internal coin flips of KGen and $\mathcal{F}$, and the choices of m only for the uuf-koa game. On the relationship between these two security notions, the following proposition holds.

Proposition 1 ([GMR88]). *Let* Sig *be a signature scheme, and let Q_s be a polynomial in a security parameter λ. If there exists a PPT forger algorithm which wins the uuf-koa game of* Sig*, then there exists a PPT forger algorithm which wins the Q_s-euf-cma game of* Sig.

2.2 Cryptographic Assumption

We now introduce the One-More DL (OM-DL) assumption. Let ℓ_p be a polynomial in λ. We write GGen to denote a PPT group parameter generator which takes a security parameter 1^λ as input, and then outputs a *group parameter* $(\mathbb{G}, p, g)$ of a group description $\mathbb{G}$ which is of prime order p such that $p < 2^{\ell_p}$ with a generator g. For any group parameter $(\mathbb{G}, p, g) \leftarrow \mathsf{GGen}(1^\lambda)$ and any element $y \in \mathbb{G}$, an element $x \in \mathbb{Z}_p$ is said to be the *discrete logarithm (DL) of y* if it holds that $y = g^x$ in $\mathbb{G}$.

Let T be a polynomial in λ. An algorithm $\mathcal{A}$ is said to *solve the T-OM-DL problem* if a challenger C outputs 1 in the *T-OM-DL game* that is defined in the following way: on a security parameter λ,

OM Init $\mathcal{A}$ is given a tuple $(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$ where C generates a group parameter $(\mathbb{G}, p, g) \leftarrow \mathsf{GGen}(1^\lambda)$, and then samples $T + 1$ distinct instances $y_1, \ldots, y_{T+1} \in_{\mathrm{U}} \mathbb{G}$.
DL Oracle $\mathcal{A}$ is allowed to access the *DL oracle*. Namely, when $\mathcal{A}$ sends a t-th query $\overline{y}_t \in \mathbb{G}$, $\mathcal{A}$ receives the DL $\overline{x}_t \in \mathbb{Z}_p$ of $\overline{y}_t$.
OM Challenge When $\mathcal{A}$ eventually outputs a tuple $(x_1, x_2, \ldots, x_{T+1})$, C outputs 1 if $\mathcal{A}$ made at most T queries to the DL oracle in **DL Oracle** phase, and for any $1 \le t \le T + 1$, x_t is the DL of y_t.

The *T-OM-DL assumption* states that any PPT algorithm $\mathcal{A}$ solves the T-OM-DL problem with negligible probability.

3 Impossibility on Schnorr Signature in NPROM

In this section, we show the impossibility of proving that the Schnorr signature is uuf-koa from the T-OM-DL assumption in the NPROM.

3.1 Schnorr Signature

We now introduce the Schnorr signature [Sch91].

$\mathsf{KGen}(1^\lambda)$ outputs (sk, pk), where $(\mathbb{G}, p, g) \leftarrow \mathsf{GGen}(1^\lambda)$, $sk \in_{\mathrm{U}} \mathbb{Z}_p$, $y := g^{sk}$, and $pk := (\mathbb{G}, p, g, y)$.
$\mathsf{Sign}(sk, pk, m)$ outputs a signature $\sigma := (\mathrm{cmt}, \mathrm{res})$ on the message $m \in \{0,1\}^{\ell_{\mathsf{m}}}$, where ℓ_{m} is a polynomial in λ, under the public key pk. The procedure is as follows:
(1) $\mathrm{st} \in_{\mathrm{U}} \mathbb{Z}_p$ and then $\mathrm{cmt} := g^{\mathrm{st}}$,
(2) $\mathrm{cha} := H(\mathrm{cmt}, m)$, where $H : \{0,1\}^* \to \{0,1\}^{2\ell_p}$,
(3) $\mathrm{res} := \mathrm{st} + sk \cdot \mathrm{cha}$.
$\mathsf{Ver}(pk, m, \sigma)$ outputs 1 if we have $\mathrm{cmt} = g^{\mathrm{res}} y^{-H_{pk}(\mathrm{cmt}, m)}$.

Note that we consider the hash function $H : \{0,1\}^* \to \{0,1\}^{2\ell_p}$ instead of $H' : \{0,1\}^* \to \mathbb{Z}_p$. This is because the uniform distribution over $\{0,1\}^{2\ell_p}$ can be seen as the one over $\mathbb{Z}_p$ regardless of the order p by the following lemma.

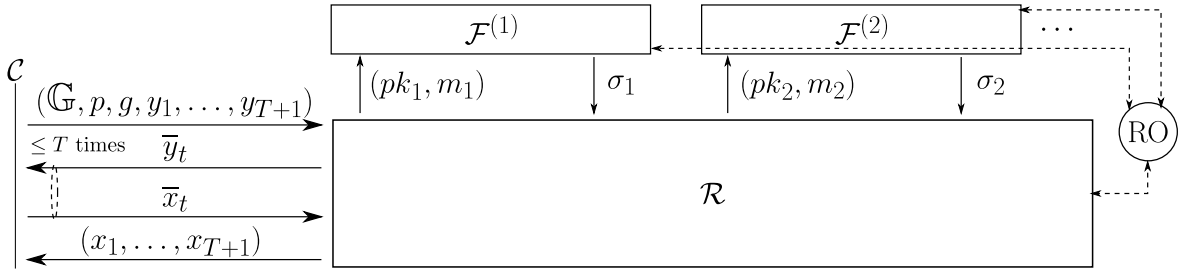


Fig. 1. The overview of a reduction $\mathcal{R}(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$

Lemma 2. *Let ℓ_p be a polynomial in λ, and let $p < 2^{\ell_p}$. Then, the distribution of $z \bmod p$ for $z \in_{\mathrm{U}} \{0,1\}^{2\ell_p}$ is statistically close to $U(\mathbb{Z}_p)$.*

Proof. To prove this lemma, we use the following fact.

Lemma 3 ([FHIS14, Lemma 3]). *Let $\{n_1(\lambda)\}_\lambda$ and $\{n_2(\lambda)\}_\lambda$ be two sequences of natural numbers. Assume that n_2/n_1 is negligible in λ. If $z \in_{\mathrm{U}} \mathbb{Z}_{n_1}$, then the distribution of $z \bmod n_2$ is statistically close to the uniform distribution over $\mathbb{Z}_{n_2}$.*

Since $p/2^{2\ell_p} < 2^{\ell_p}/2^{2\ell_p} = 1/2^{\ell_p}$ is negligible in λ, Lemma 3 implies that the distribution of $z \bmod p$ for $z \in_{\mathrm{U}} \{0,1\}^{2\ell_p}$ is statistically close to $U(\mathbb{Z}_p)$. □

We note the notation $H(\mathrm{cmt}, m)$. The string cha is defined as the hash value of $H(\mathrm{cmt}, m)$. The domain of the hash function H is defined as $\{0,1\}^*$, and the input pair (cmt, m) is in $\mathbb{G} \times \{0,1\}^{\ell_m}$. We consider that the hash value $H(\mathrm{cmt}, m)$ is computed on the concatication of the binary representation of $\mathrm{cmt} \in \mathbb{G}$ and $m \in \{0,1\}^{\ell_m}$.

The Schnorr signature is known to have the honest-verifier zero knowledge property. Namely, there exists a PPT simulator which takes a public key $pk = (\mathbb{G}, p, g, y)$ and a string $\mathrm{cha} \in \{0,1\}^{2\ell_p}$ as input, and then return $(\mathrm{cmt}, \mathrm{cha}, \mathrm{res})$ such that $\mathsf{Ver}(pk, m, \sigma) = 1$. Moreover, for any $(pk, sk) \leftarrow \mathsf{KGen}(1^\lambda)$ and any $m \in \{0,1\}^{\ell_m}$, the distribution of $(\mathrm{cmt}, \mathrm{res})$ is identical to that of $\left(\overline{\mathrm{cmt}}, \overline{\mathrm{res}}\right) \leftarrow \mathsf{Sign}(sk, pk, m)$ if the distribution of cha given to the simulator coincides with that of $H(\overline{\mathrm{cmt}}, m)$. Note that the above property follows in the random oracle model by sampling a string cha uniformly at random from $\{0,1\}^{2\ell_p}$, since $H(\overline{\mathrm{cmt}}, m)$ is uniformly distributed over $\{0,1\}^{2\ell_p}$ in the random oracle model.

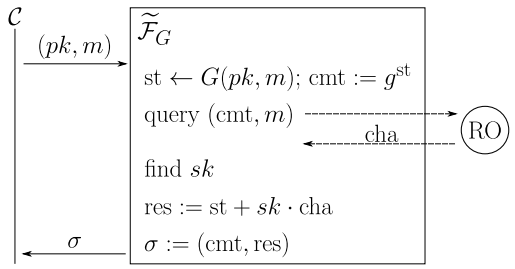
3.2 Our Impossibility Result

Let T be a polynomial in λ. We now explain the situation where *the Schnorr signature is provable to be uuf-koa from the T-OM-DL assumption.* This is defined by the manner of the black-box reduction such as [PS00,FF13]. Namely, there exist a non-negligible function ϵ and a PPT reduction algorithm $\mathcal{R}$ such that $\mathcal{R}$ solves the T-OM-DL problem with probability ϵ by invoking a forger $\mathcal{F}$ which wins the uuf-koa game. Here, $\mathcal{R}$ is allowed to access the DL oracle at most T times, since $\mathcal{R}$ aims to win the T-OM-DL game.

Let $(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$ be a T-OM-DL instance given from the T-OM-DL challenger C to the reduction $\mathcal{R}$. $\mathcal{R}$ aims to find the solution $(x_1, x_2, \ldots, x_{T+1})$ of the instance $(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$. For this purpose, $\mathcal{R}$ would access the DL oracle at most T times and invoke a uuf-koa forger $\mathcal{F}$ polynomially many times. $\mathcal{R}$ plays a role of a T-OM-DL adversary in the T-OM-DL game, and plays a uuf-koa challenger in the uuf-koa game simultaneously. On a t-th DL oracle query, $\mathcal{R}$ sends a t-th instance $\overline{y}_t \in \mathbb{G}$ to receive its DL $\overline{x}_t \in \mathbb{Z}_p$. On the other hand, $\mathcal{R}$ invokes a forger $\mathcal{F}$ of the Schnorr signature. Suppose that $\mathcal{R}$ invokes $\mathcal{F}$ with a pair (pk, m). Then, $\mathcal{F}$ returns a *forgery* σ, namely a valid signature σ of m.

We consider the security in the *non-programmable random oracle model (NPROM)* [FF13]. In the NPROM, $\mathcal{R}$ and $\mathcal{F}$ should obtain hash values from the random oracle in a similar manner to the ordinary ROM. However, the random oracle is dealt with an independent party from $\mathcal{R}$ and $\mathcal{F}$ in the security proof. This means that $\mathcal{R}$ is prohibited to simulate the random oracle internally, whereas such a simulation is allowed in the ordinary ROM. Hence, $\mathcal{R}$ can observe all random oracle queries by $\mathcal{F}$, but it is not allowed to program these values.

$\mathcal{R}$ is allowed to concurrently and adaptively invoke $\mathcal{F}$ at most I times and rewind it polynomially many times for some polynomial I. $\mathcal{R}$ would eventually behave as follows. On a T-OM-DL instance $(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$ to $\mathcal{R}$, $\mathcal{R}$ would execute the following processes concurrently:



(1) $\mathrm{st} \leftarrow G(pk, m)$; $\mathrm{cmt} := g^{\mathrm{st}}$.
(2) query (cmt, m) to the random oracle.
(3) after obtaining the hash value cha of (cmt, m), find the secret key sk of pk.
(4) $\mathrm{res} := \mathrm{st} + sk \cdot \mathrm{cha}$.
(5) return $\sigma = (\mathrm{cmt}, \mathrm{res})$.

Fig. 2. uuf-koa forger $\widetilde{\mathcal{F}}_G(pk, m)$, where $pk = (\mathbb{G}, p, g, y)$

Access to DL oracle When $\mathcal{R}$ sends a t-th instance $\overline{y}_t$ to the DL oracle, it receives its DL $\overline{x}_t$.

Request to obtain the hash value When $\mathcal{R}$ makes a i-th pair $\left(\overline{\mathrm{cmt}}, \overline{m}\right)$ to the random oracle oracle, it receives its hash value $\overline{\mathrm{cha}}$.

Invocation of $\mathcal{F}$ When $\mathcal{R}$ invokes a k-th forger $\mathcal{F}^{(k)}$ on (pk_k, m_k), $\mathcal{R}$ obtains a forgery $\sigma_k := (\mathrm{cmt}_k, \mathrm{res}_k)$ after $\mathcal{F}^{(k)}$ obtains the hash value cha_k of the pair (cmt_k, m_k).

Finally, $\mathcal{R}$ outputs the solution $(x_1, x_2, \ldots, x_{T+1})$ of $(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$ with probability ϵ. The behavior of $\mathcal{R}$ are depicted as in Fig. 1. Note that $\mathcal{R}$ may rewind $\mathcal{F}^{(k)}$ just after $\mathcal{F}^{(k)}$ queries to the random oracle. This is because $\mathcal{R}$ can observe the queries and the responses of $\mathcal{F}^{(k)}$ in the NPROM setting.

For the input $pk_k = (\mathbb{G}_k, p_k, g_k, y_k)$ to $\mathcal{F}$ from $\mathcal{R}$, we assume only that the length of the order p_k is ℓ_p, unlike the key-preserving reduction [PV05,FH18] which requires that each pk_k must coincide the T-OM-DL instance given to $\mathcal{R}$.

We now show the impossibility of the provable security of the Schnorr signature in the NPROM.

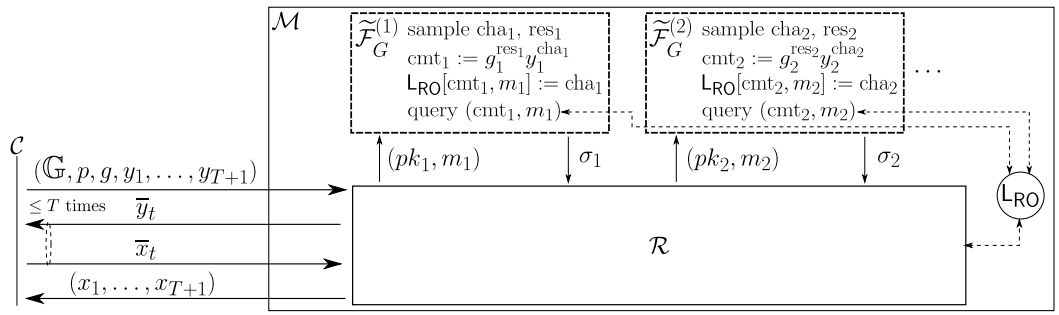
Theorem 4. *Let ℓ_p, T, and I be polynomials in λ, and let ϵ be a non-negligible function. Assume that there exist a PPT reduction algorithm $\mathcal{R}$ which solves the T-OM-DL problem with probability ϵ by invoking a forger $\mathcal{F}$ at most I times such that $\mathcal{F}$ wins the uuf-koa game on a public key $pk_k = (\mathbb{G}_k, p_k, g_k, y_k)$ for $p_k < 2^{\ell_p}$ and $g_k, y_k \in \mathbb{G}_k$. Then there exists a PPT algorithm $\mathcal{M}$ which solves the T-OM-DL problem with probability $\epsilon -$ negl.*

Proof. Assume that there exists a PPT Turing reduction algorithm $\mathcal{R}$ which solves the T-OM-DL problem with probability ϵ by invoking a forger $\mathcal{F}$ that wins the uuf-koa game at most I times. As mentioned above, $\mathcal{R}$ can find the solution $(x_1, \ldots, x_{T+1})$ of a given T-OM-DL instance $(\mathbb{G}, p, g, y_1, \ldots, y_{T+1})$ with probability ϵ if a forger $\mathcal{F}$ which can win the uuf-koa game is provided to $\mathcal{R}$ no matter what $\mathcal{F}$ executes. We first describe a specific hypothetical uuf-koa forgers $\widetilde{\mathcal{F}}$. We shall next construct a meta-reduction algorithm $\mathcal{M}$ which solves the T-OM-DL problem with non-negligible probability by utilizing $\mathcal{R}$ and simulating $\widetilde{\mathcal{F}}$.

Family $\left\{\widetilde{\mathcal{F}}_G\right\}_G$ of Hypothetical Forgers We consider a family of hypothetical uuf-koa forgers $\widetilde{\mathcal{F}}_G$ which are parameterized by a deterministic random function $G : \{0,1\}^* \to \{0,1\}^{2\ell_p}$. $\widetilde{\mathcal{F}}_G$ aims to return a forgery $\sigma = (\mathrm{cmt}, \mathrm{res})$ whose distribution is statistically close to that of $\mathsf{Sign}(sk, pk, m)$. The formal description of $\widetilde{\mathcal{F}}_G$ is given in Fig 2. Since we now consider the security game in the NPROM, all hash values are supposed to be obtained from the external random oracle. By Lemma 2, the distribution of st generated by the random function G in the process (1) is statistically close to the uniform distribution over $\mathbb{Z}_p$. Then, we have that $\mathcal{F}$ generates $(\mathrm{cmt}, \mathrm{cha}, \mathrm{res})$ whose distribution is statistically close to that of $\mathsf{Sign}(sk, pk, m)$ by the description from $\mathcal{F}$. Thus $\widetilde{\mathcal{F}}_G$ can always win the uuf-koa game, and the distribution of the forgery σ is statistically close to that of honestly generated signatures.

It should be noted that the process (3) seems not to be done in PPT, however, we will construct a meta-reduction $\mathcal{M}$ which simulates the forger $\widetilde{\mathcal{F}}_G$ for $\mathcal{R}$ in PPT.

We fix a function G, and consider that the reduction $\mathcal{R}$ invokes the forger $\widetilde{\mathcal{F}}_G$ above at most I times. For any $1 \le k \le I$, we explicitly denote by $\widetilde{\mathcal{F}}_G^{(k)}$ the hypothetical forger $\widetilde{\mathcal{F}}_G$ which is invoked at k-th time. $\mathcal{R}$ may rewind some k-th invocation $\widetilde{\mathcal{F}}_G^{(k)}$. However, $\widetilde{\mathcal{F}}_G^{(k)}$ is deterministic, since st is fixed by G, any hash value is determined by the random oracle, and pk has only one secret key. Namely, the behavior of $\widetilde{\mathcal{F}}_G^{(k)}$ is identical for the same inputs.



– sample random coins r and initialize L_{RO} and L_G. Then run $\mathcal{R}(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1}; r)$. During that, proceed to the following according to the $\mathcal{R}$'s output:
Accessing DL oracle with $\overline{y}_t \in \mathbb{G}$: forward $\overline{y}_t$ to DL oracle, and then reply $\overline{x}_t$ obtained from DL oracle.
Requesting the hash value of $\left(\overline{\mathrm{cmt}}, \overline{m}\right)$: return $\overline{\mathrm{cha}} \leftarrow\!\!\!\sim_{U(\{0,1\}^{2\ell_p})} \mathsf{L}_{\mathsf{RO}}\left[\overline{\mathrm{cmt}}, \overline{m}\right]$.
Invoking a k-th forger $\widetilde{\mathcal{F}}_G^{(k)}$ on (pk_k, m_k): start to run the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$, where the simulator is defined as follows:
(i) $\mathrm{cha}_k \leftarrow\!\!\!\sim_{U(\{0,1\}^{2\ell_p})} \mathsf{L}_G\,[pk_k, m_k]$.
(ii) $\mathrm{res}_k \leftarrow\!\!\!\sim_{U(\{0,1\}^{2\ell_p})} \mathsf{L}_G\,[pk_k, m_k, \mathrm{cha}_k]$.
(iii) $\mathrm{cmt}_k := g_k^{\mathrm{res}_k} y_k^{-\mathrm{cha}_k}$, where $pk_k = (\mathbb{G}_k, p_k, g_k, y_k)$.
(iv) if $\mathsf{L}_{\mathsf{RO}}\,[\mathrm{cmt}_k, m_k] \notin \{\mathrm{cha}_k, \perp\}$, then abort.
(v) $\mathsf{L}_{\mathsf{RO}}\,[\mathrm{cmt}_k, m_k] := \mathrm{cha}_k$.
(vi) query (cmt_k, m_k) to the random oracle.
(vii) after obtaining the hash value cha_k, return $\sigma_k := (\mathrm{cmt}_k, \mathrm{res}_k)$.
– When $\mathcal{R}$ outputs a tuple $(x_1, x_2, \ldots, x_{T+1})$, output it and then halt.

Fig. 3. Meta-reduction $\mathcal{M}(\mathbb{G}, p, g, y_1, y_2, \ldots, y_{T+1})$

Meta-Reduction $\mathcal{M}$ We depict the meta-reduction $\mathcal{M}$ in Fig. 3. We explain the idea of constructing the meta-reduction $\mathcal{M}$. $\mathcal{M}$ aims to make $\mathcal{R}$ to solve the T-OM-DL problem. Recall that $\mathcal{R}$ can solve the T-OM-DL problem with non-negligible probability ϵ if $\widetilde{\mathcal{F}}_G$ is provided. The main point to construct $\mathcal{M}$ is how to simulate $\widetilde{\mathcal{F}}_G$ for $\mathcal{R}$. We now fix a k-th invocation $\widetilde{\mathcal{F}}_G^{(k)}$ on a pair (pk_k, m_k). $\mathcal{M}$ is required to return a forgery, although this task seems not to be in PPT in general. To overcome the difficulty, $\mathcal{M}$ utilizes the honest-verifier zero-knowledge property of the Schnorr signature and the ability of simulating the random oracle which is allowed in the NPROM-PROM model. In other words, $\mathcal{M}$ generates $(\mathrm{cmt}_k, \mathrm{cha}_k, \mathrm{res}_k)$ in the same way as the simulator which is assumed in the honest-verifier zero-knowledge property does. In the forgery generation, the secret key of pk is no longer needed. Then, $\mathcal{M}$ programs cha_k as the hash value of the pair (cmt_k, m_k). Thus, $\mathcal{M}$ succeeds in simulating $\widetilde{\mathcal{F}}_G^{(k)}$ by returning $\sigma_k = (\mathrm{cmt}_k, \mathrm{res}_k)$ as a forgery.

We now show that $\mathcal{M}$ can solve the T-OM-DL problem by ensuring that $\mathcal{M}$ correctly simulates $\widetilde{\mathcal{F}}_G^{(k)}$. This is proven by the following lemmas.

Lemma 5. *We fix $1 \leq k \leq I$. Assume that the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ by $\mathcal{M}$ does not abort. The distribution of the output by the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ by $\mathcal{M}$ is identical to the one by the hypothetical forger $\widetilde{\mathcal{F}}_G^{(k)}$, which is given a public key $pk_k = (\mathbb{G}_k, p_k, g_k, y_k)$ and a message m_k, if $p_k < 2^{\ell_p}$.*

Proof. We fix an index k. Assume that the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ by $\mathcal{M}$ does not abort. First, cha_k of both the simulator and the hypothetical forger are uniformly distributed over $\{0,1\}^{2\ell_p}$, since cha_k is obtained from the random oracle in both case.

We now consider a map $\tau_{h,a}$ which maps $x \in \mathbb{Z}_{p_k}$ to $\left(g_k^x \cdot h, x + a\right)$ for any $h \in \mathbb{G}_k$ and any $a \in \mathbb{Z}_{p_k}$. Observe that $\tau_{h,a}$ is bijective, since g_k is a generator of $\mathbb{G}_k$. Lemma 2 and $p_k < 2^{\ell_p}$ imply that for any $h \in \mathbb{G}_k$ and any $a \in \mathbb{Z}_{p_k}$, the distribution of $\tau_{h,a}(x \bmod p_k)$ is statistically close to $U\left(\mathbb{G}_k \times \mathbb{Z}_{p_k}\right)$, if $x \in_{\mathrm{U}} \{0,1\}^{2\ell_p}$. Then, $(\mathrm{cmt}_k, \mathrm{res}_k)$ of the simulator can be represented as $\tau_{y_k^{-\mathrm{cha}_k},0}(\mathrm{res}_k)$, whereas the one of the hypothetical forger can be represented as $\tau_{e_k, sk_k \cdot \mathrm{cha}_k}(\mathrm{st}_k)$, where e_k is the identity of $\mathbb{G}_k$. It follows that the distribution of $(\mathrm{cmt}_k, \mathrm{res}_k)$ of the simulator coincides with that of the hypothetical forger. Thus, the distributions of the output $(m_k, \mathrm{cmt}_k, \mathrm{cha}_k, \mathrm{res}_k)$ by both forgers are identical. □

Lemma 6. *For any $1 \leq k \leq I$, the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$, which is given $pk_k = (\mathbb{G}_k, p_k, g_k, y_k)$ and m_k, aborts with probability $2/p_k + \mathrm{negl}$.*

Proof. We fix an index k. The simulator $\widetilde{\mathcal{F}}_G^{(k)}$ aborts when the event $\mathsf{L}_{\mathsf{RO}}[\mathrm{cmt}_k, m_k] \notin \{\mathrm{cha}_k, \perp\}$ occurs. The event $\mathsf{L}_{\mathsf{RO}}[\mathrm{cmt}_k, m_k] \notin \{\mathrm{cha}_k, \perp\}$ means that the hash value of (cmt_k, m_k) is already defined as the different value from cha_k before the process (iv) of the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$. One can consider the two cases that the hash value of (cmt_k, m_k) is defined before the process (iv).

The first one is that $\mathcal{R}$ finds cmt_k and then makes the query (cmt_k, m_k) to the random oracle before cmt_k is output by $\mathcal{M}$. As estimated in the proof of Lemma 5, the distribution of cmt_k set by the simulator at (iii) is statistically close to $U(\mathbb{G}_k)$. This implies that the probability that $\mathcal{R}$ can find cmt_k before it is given from $\mathcal{M}$ is $1/p_k + \mathrm{negl}$.

The second one is that for some $k' < k$, the simulator of $\widetilde{\mathcal{F}}_G^{(k')}$ defines it at (v). We focus on the situation where $\mathcal{R}$ invokes $\widetilde{\mathcal{F}}_G^{(k')}$ with $(pk_k, m_k) = (pk_{k'}, m_{k'})$. Observe that $(\mathrm{cmt}_{k'}, \mathrm{cha}_{k'}, \mathrm{res}_{k'}) = (\mathrm{cmt}_k, \mathrm{cha}_k, \mathrm{res}_k)$, because $\mathrm{cha}_{k'}$, cha_k, $\mathrm{res}_{k'}$ and res_k are lazily sampled. This implies that cha_k is already defined as the hash value of (cmt_k, m_k). Thus, the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ does not abort. We next focus on the opposite situation. Namely, $\mathcal{R}$ invokes $\widetilde{\mathcal{F}}_G^{(k)}$ with $(pk_k, m_k) \neq (pk_{k'}, m_{k'})$. In this situation, the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ aborts if the binary representation of (cmt_k, m_k) coincides with that of $(\mathrm{cmt}_{k'}, m_{k'})$. Since the distribution of cmt_k chosen in the process (iii) is statistically close to $U(\mathbb{G}_k)$ for any invocation, the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ sets cmt_k so that the representation of $\mathrm{cmt}_{k'}$ coincides with cmt_k with the probability at most $1/p_k + \mathrm{negl}$. Thus, $\mathcal{M}$ aborts in the process (iv) with $2/p_k + \mathrm{negl}$ for the k-th invocation $\widetilde{\mathcal{F}}_G^{(k)}$. □

It follows from Lemma 5 that $\mathcal{M}$ can make $\mathcal{R}$ to return the solution of the given T-OM-DL instance, if $\mathcal{M}$ does not abort in the simulation of $\widetilde{\mathcal{F}}_G^{(k)}$. This is because the behavior of the simulator of $\widetilde{\mathcal{F}}_G^{(k)}$ by $\mathcal{M}$ is identical to that of the hypothetical one $\widetilde{\mathcal{F}}_G^{(k)}$. On the other hand, for each $1 \leq k \leq I$, Lemma 6 and $p_k < 2^{\ell_p}$ imply that $\mathcal{M}$ aborts with the probability at most $2/p_k + \mathrm{negl} > 2/2^{\ell_p} + \mathrm{negl}$ on the simulation of $\widetilde{\mathcal{F}}_G^{(k)}$. By letting $P = 2/2^{\ell_p} + \mathrm{negl}$, the abort probability of $\mathcal{M}$ is evaluated by $1 - \prod_{k=1}^{I}(1 - (2/p_k + \mathrm{negl})) < 1 - \prod_{k=1}^{I}(1 - P) = 1 - (1-P)^I$. By the binomial expansion, we have

$$
\begin{aligned}
1 - (1-P)^I &= 1 - 1 - \sum_{k=1}^{I} \binom{I}{k} (-P)^k \\
&= -\sum_{k=1}^{\lfloor I/2 \rfloor} \left(\binom{I}{2k-1} (-P)^{2k-1} + \binom{I}{2k} (-P)^{2k} \right) - (-P)^I \cdot (I \bmod 2) \\
&= \sum_{k=1}^{\lfloor I/2 \rfloor} \left(\binom{I}{2k-1} P^{2k-1} - \binom{I}{2k} P^{2k} \right) + P^I \cdot (I \bmod 2) \\
&\leq \sum_{k=1}^{\lfloor I/2 \rfloor} \binom{I}{2k-1} P^{2k-1} + P^I \\
&= \sum_{k=1}^{\lfloor I/2 \rfloor} \frac{I!}{(2k-1)!(I-2k+1)!} P^{2k-1} + P^I \\
&= \sum_{k=1}^{\lfloor I/2 \rfloor} \frac{\prod_{i=1}^{2k-1} (I-2k+1+i)}{(2k-1)!} P^{2k-1} + P^I \\
&\leq \sum_{k=1}^{\lfloor I/2 \rfloor} \left(\prod_{i=1}^{2k-1} I \right) P^{2k-1} + P^I \\
&= \sum_{k=1}^{\lfloor I/2 \rfloor} (IP)^{2k-1} + P^I.
\end{aligned}
$$

Since I is polynomial and P is negligible, it holds that $0 < IP < 1$. Therefore, we have

$$
\sum_{k=1}^{\lfloor I/2 \rfloor} (IP)^{2k-1} + P^I \leq \sum_{k=1}^{\lfloor I/2 \rfloor} IP + P^I \leq \frac{I^2}{2} P + P^I
$$

Then the success probability of $\mathcal{M}$ is at least $\epsilon - \left(\frac{I^2}{2} P + P^I \right) = \epsilon - \mathrm{negl}$. Observe that $\mathcal{M}$ runs in polynomial-time. Thus, the PPT algorithm $\mathcal{M}$ can solve the T-OM-DL game with probability $\epsilon - \mathrm{negl}$. □

The following is shown by Theorem 4 and Proposition 1.

Corollary 7. *Let ℓ_p, T, I, and Q_s be polynomials in λ, and let ϵ be a non-negligible function. Assume that there exist a PPT reduction algorithm $\mathcal{R}$ which solves the T-OM-DL problem with probability ϵ by invoking a forger $\mathcal{F}$ at most I times such that $\mathcal{F}$ wins the Q_s-euf-cma game on a public key $pk_k = (\mathbb{G}_k, p_k, g_k, y_k)$ for $p_k < 2^{\ell_p}$ and $g_k, y_k \in \mathbb{G}_k$. Then there exists a PPT algorithm $\mathcal{M}$ which solves the T-OM-DL problem with probability $\epsilon - \mathrm{negl}$.*

Proof. Assume that there exist a PPT reduction algorithm $\mathcal{R}$ which solves the T-OM-DL problem with probability ϵ by invoking a forger $\mathcal{F}$ that wins the Q_s-euf-cma game at most I times. Proposition 1 implies that $\mathcal{R}$ can solve the T-OM-DL problem with probability ϵ even when a forger $\mathcal{F}$ that wins the uuf-koa game is provided. Then it follows from Theorem 4 the construction of a PPT algorithm $\mathcal{M}$ which solves the T-OM-DL problem with probability $\epsilon - \mathrm{negl}$. □

4 Concluding Remarks

In this paper, we have shown that the Schnorr signature is unprovable to be universally unforgeable against the key-only attack (uuf-koa) from the OM-DL assumption in the NPROM via a Turing reduction as long as the OM-DL assumption holds. We have also discussed that our impossibility result for the uuf-koa security implies the impossibility of the ordinary euf-cma security.

Our result is shown by using the meta-reduction technique [BV98]. Namely, we have constructed a meta-reduction $\mathcal{M}$ which solves the OM-DL problem with the help of an assumed reduction $\mathcal{R}$ which solves the OM-DL problem with black-box access to a uuf-koa forger $\mathcal{F}$ of the Schnorr signature. In the proof of the result, we employ the NPROM-PROM model [FHJ20] which allows a meta-reduction to simulate the external random oracle. It remains open whether or not the same impossibility result holds in other model such as the conventional NPROM-NPROM model.

Acknowledgment

We would like to thank anonymous reviewers for their valuable comments and suggestions to the previous versions. This work was supported in part by JSPS KAKENHI Grant Numbers JP18K11288 and JP19K20272.

References

[AABN08] Michel Abdalla, Jee Hea An, Mihir Bellare, and Chanathip Namprempre. From identification to signatures via the Fiat-Shamir transform: Necessary and sufficient conditions for security and forward-security. *Information Theory, IEEE Transactions on*, 54(8):3631–3646, 2008.

[BJLS16] Christoph Bader, Tibor Jager, Yong Li, and Sven Schäge. On the impossibility of tight cryptographic reductions. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016*, volume 9666 of *LNCS*, pages 273–304. Springer, Heidelberg, 2016.

[BMV08] Emmanuel Bresson, Jean Monnerat, and Damien Vergnaud. Separation results on the "One-more" computational problems. In Tal Malkin, editor, *CT-RSA 2008*, volume 4964 of *LNCS*, pages 71–87. Springer, Heidelberg, 2008.

[BNPS03] M. Bellare, C. Namprempre, D. Pointcheval, and M. Semanko. The One-more-RSA-inversion problems and the security of Chaum's blind signature scheme. *J. Cryptology*, 16(3):185–215, 2003.

[Bon98] Dan Boneh. The decision Diffie-Hellman problem. In Joe P. Buhler, editor, *Algorithmic Number Theory*, volume 1423 of *LNCS*, pages 48–63. Springer, Heidelberg, 1998.

[BP02] Mihir Bellare and Adriana Palacio. GQ and Schnorr identification schemes: Proofs of security against impersonation under active and concurrent attacks. In Moti Yung, editor, *EUROCRYPT 2002*, volume 2442 of *LNCS*, pages 162–177. Springer, Heidelberg, 2002.

[BV98] Dan Boneh and Ramarathnam Venkatesan. Breaking RSA may not be equivalent to factoring. In Kaisa Nyberg, editor, *EUROCRYPT'98*, volume 1403 of *LNCS*, pages 59–71. Springer, Heidelberg, 1998.

[Cor02] Jean-Sébastien Coron. Optimal security proofs for pss and other signature schemes. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 272–287. Springer, Heidelberg, 2002.

[FF13] Marc Fischlin and Nils Fleischhacker. Limitations of the meta-reduction technique: The case of Schnorr signatures. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 444–460. Springer, Heidelberg, 2013.

[FH16] Masayuki Fukumitsu and Shingo Hasegawa. Impossibility on the provable security of the Fiat-Shamir-type signatures in the non-programmable random oracle model. In M. Bishop and A.C.A. Nascimento, editors, *ISC 2016*, volume 9866 of *LNCS*, pages 389–407. Springer, Heidelberg, 2016.

[FH17] Masayuki Fukumitsu and Shingo Hasegawa. Impossibility of the provable security of the Schnorr signature from the One-More DL assumption in the non-programmable random oracle model. In Tatsuaki Okamoto, Yong Yu, Man Ho Au, and Yannan Li, editors, *ProvSec 2017*, volume 10592 of *LNCS*, pages 201–218. Springer, Heidelberg, 2017.

[FH18] Masayuki Fukumitsu and Hasegawa. Black-box separations on Fiat-Shamir-type signatures in the non-programmable random oracle model. *IEICE Trans. Fundamentals, Special Section on Cryptography and Information Security*, E101-A(1):77–87, 2018.

[FH20] Masayuki Fukumitsu and Shingo Hasegawa. One-more assumptions do not help Fiat-Shamir-type signature schemes in nprom. In Stanislaw Jarecki, editor, *Topics in Cryptology – CT-RSA 2020*, pages 586–609, Cham, 2020. Springer International Publishing.

[FHIS14] Masayuki Fukumitsu, Shingo Hasegawa, Shuji Isobe, and Hiroki Shizuya. The RSA group is adaptive pseudo-free under the RSA assumption. *IEICE Trans. Fundamentals, Special Section on Cryptography and Information Security*, E97-A(1):200–214, 2014.

[FHJ20] Marc Fischlin, Patrick Harasser, and Christian Janson. Signatures from sequential-or proofs. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology – EUROCRYPT 2020*, pages 212–244, Cham, 2020. Springer International Publishing.

[FJS14] Nils Fleischhacker, Tibor Jager, and Dominique Schröder. On tight security proofs for Schnorr signatures. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014*, volume 8873 of *LNCS*, pages 512–531. Springer, Heidelberg, 2014.

[FLR+10] Marc Fischlin, Anja Lehmann, Thomas Ristenpart, Thomas Shrimpton, Martijn Stam, and Stefano Tessaro. Random oracles with(out) programmability. In Masayuki Abe, editor, *ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 303–320. Springer, Heidelberg, 2010.

[FPS20] Georg Fuchsbauer, Antoine Plouviez, and Yannick Seurin. Blind schnorr signatures and signed elgamal encryption in the algebraic group model. In Anne Canteaut and Yuval Ishai, editors, *Advances in Cryptology – EUROCRYPT 2020*, pages 63–95, Cham, 2020. Springer International Publishing.

[FS87] Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO'86*, volume 263 of *LNCS*, pages 186–194. Springer, Heidelberg, 1987.

[GBL08] Sanjam Garg, Raghav Bhaskar, and SatyanarayanaV. Lokam. Improved bounds on security reductions for discrete log based signatures. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 93–107. Springer, Heidelberg, 2008.

[GMR88] Shafi Goldwasser, Silvio Micali, and Ronald L Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comput.*, 17(2):281–308, 1988.

[KK12] Saqib A. Kakvi and Eike Kiltz. Optimal security proofs for full domain hash, revisited. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 537–553. Springer, Heidelberg, 2012.

[Nie02] JesperBuus Nielsen. Separating random oracle proofs from complexity theoretic proofs: The non-committing encryption case. In Moti Yung, editor, *CRYPTO 2002*, volume 2442 of *LNCS*, pages 111–126. Springer, Heidelberg, 2002.

[Pas11] Rafael Pass. Limits of provable security from standard assumptions. In *STOC2011*, pages 109–118, 2011.

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

[PV05] Pascal Paillier and Damien Vergnaud. Discrete-log-based signatures may not be equivalent to discrete log. In Bimal Roy, editor, *ASIACRYPT 2005*, volume 3788 of *LNCS*, pages 1–20. Springer, Heidelberg, 2005.

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

[Seu12] Yannick Seurin. On the exact security of Schnorr-type signatures in the random oracle model. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 554–571. Springer, Heidelberg, 2012.

[ZCC+15] Zongyang Zhang, Yu Chen, Sherman S. M. Chow, Goichiro Hanaoka, Zhenfu Cao, and Yunlei Zhao. Black-box separations of hash-and-sign signatures in the non-programmable random oracle model. In Man-Ho Au and Atsuko Miyaji, editors, *Provable Security 2015*, volume 9451 of *LNCS*, pages 435–454. Springer, Heidelberg, 2015.

[ZZC+14] Jiang Zhang, Zhenfeng Zhang, Yu Chen, Yanfei Guo, and Zongyang Zhang. Black-box separations for One-more (static) CDH and its generalization. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014*, volume 8874 of *LNCS*, pages 366–385. Springer, Heidelberg, 2014.