Separating Short Structure-Preserving Signatures from Non-Interactive Assumptions

Masayuki Abe, Jens Groth*, and Miyako Ohkubo

NTT Information Sharing Platform Laboratories
NTT Corporation
abe.masayuki@lab.ntt.co.jp

University College London, U.K.
j.groth@ucl.ac.uk

Security Architecture Laboratory, NSRI
NICT
m.ohkubo@nict.go.jp

Abstract. Structure-preserving signatures are signatures whose public keys, messages, and signatures are all group elements in bilinear groups, and the verification is done by evaluating pairing product equations. It is known that any structure-preserving signature in the asymmetric bilinear group setting must include at least 3 group elements per signature and a matching construction exists.
In this paper, we prove that optimally short structure preserving signatures cannot have a security proof by an algebraic reduction that reduces existential unforgeability against adaptive chosen message attacks to any non-interactive assumptions. Towards this end, we present a handy characterization of signature schemes that implies the separation.

Keywords. Structure-Preserving Signatures, Algebraic Reduction, Meta-Reduction

1 Introduction

1.1 Background

When messages, signatures, and verification keys are elements of bilinear groups and the signature verification is done by evaluating pairing product equations, a signature scheme is called *structure-preserving* [2]. A structure-preserving signature (SPS for short) blends well with the Groth-Sahai non-interactive proof system [24], and enables the construction of efficient cryptographic protocols such as round-optimal blind signatures [4, 2], traceable signatures [1], group encryption [10], proxy signatures [2], and delegatable credential systems [17].

* Supported by EPSRC grant number EP/G013829/1.

The first SPS was presented in [23] as a feasibility result. A variation of the Camenisch-Lysyanskaya signature scheme [9] introduced in [22] is an SPS that is secure against random message attacks. Schemes in [10] and [16] are efficient when signing a single group element, but their signature size grows linearly in the size of the message. The scheme in [16] is called automorphic as the message space includes its own public key, which is a useful feature in many applications. [2] presented the first constant-size SPS whose signature consists of 7 group elements. Yet shorter signatures have been pursued since then, however, [3] proved that any secure SPS in asymmetric bilinear groups requires at least 3 group elements. They presented a scheme matching the lower bound.

The 3-element SPS in [3] is based on a strong interactive assumption. They also constructed a 4-element SPS with a restricted message space based on a non-interactive assumption. It has been left as an open problem to find an optimal SPS based on a non-interactive assumption.

1.2 Black-box Separations

A fully black-box reduction from a primitive B to a cryptographic scheme A is an algorithm R such that for any instance f of B and for any adversary E against A, if E breaks A^f then $R^{f,E}$ breaks f. A black-box separation is to show the absence of such an algorithm R. While there are number of non-black-box techniques, e.g., [5], black-box separations are meaningful as a convincing indication of the hardness of finding a reduction and as a guide to find a way to get around it. For variations and more discussion we refer to [33].

Oracle separation and meta-reduction are widely used techniques in showing a separation. Oracle separation is useful in showing the difficulty of constructing a cryptographic scheme from a minimal primitive such as a one-way function. Since black-box reductions relativise, showing the existence of an oracle that is useful in breaking A but useless in breaking B implies absence of black-box reductions from B to A. Since the seminal work by Impagliazzo and Rudich [26], numerous results have been found using this approach. In most cases, primitives are simple cryptographic objects such as one-way functions, and the schemes in question are non-interactive ones such as collision-free hash function [34] or signature schemes [20, 14, 13]. A recent work in [27] addresses more involved interactive schemes, blind signatures, by extending this line of techniques.

In the Meta-reduction approach, initiated by [7, 11], the proof of separation is done by constructing an algorithm, a so-called meta-reduction, that uses a reduction as a black-box and solves a targeted problem, which can be the same as or different from the primitive the reduction is supposed to break. The intuition is that if a reduction is successful, the reduction breaks the underlying primitive by itself without help from the adversary. Proofs for separation exploits strong properties of the target schemes and underlying primitives. [15] exploits the blindness property in constructing a meta-reduction separating three-move blind signatures from non-interactive assumptions. In [32] a class of protocols, constant-round sequentially witness-hiding special-sound protocols for unique

witness relations, is separated from any standard assumptions. It includes some practically important protocols such as Schnorr identification schemes.

Separation is often considered for limited classes of reductions. [31] assumes a key-preserving property where the same RSA moduli are used in all oracle calls. Later in [29] an assumption so-called instance non-malleability is introduced to ease the limitation. A variation in prime-order groups appears in [28]. In [7, 11, 30, 8, 19], a class of algorithms called *algebraic reductions* is considered. In this class, yielding a new group element is limited so that it is possible to extract its representation for relevant bases. As claimed in [7], the class of algebraic reductions are not overly restrictive. In particular, for prime order groups, all known efficient reductions fall into this class to the best of our knowledge.

1.3 Our Contribution

This paper shows that no algebraic reduction falls short in proving existential unforgeability against adaptive chosen message attacks of 3-element SPS in type-III bilinear groups [18] based on *any non-interactive assumption*. This gives a partial justification for the existing 3-element schemes with interactive assumptions since algebraic algorithms, while covering all known reduction algorithms in prime order groups, are not powerful enough to prove the security of a 3-element SPS.

Our separation follows the meta-reduction paradigm. However, instead of showing a monolithic proof that constructs a meta-reduction from scratch, we present a handy characterization that separates a signature scheme from any non-interactive assumptions. It facilitates the proofs, in particular when the reductions are restricted to a class of algorithms where knowledge extraction is given for free. The intuition behind our characterization is that if the signature scheme in question forces a reduction algorithm to know some information, e.g., the signing-key itself, to simulate the signing oracle in the EUF-CMA game, and this information is so essential that the adversary wins the game by seeing it, then the reduction algorithm can break the assumption without help from the adversary. Given the characterization, we show that such *crucial information* exists in any 3-element SPS when the reduction algorithm is algebraic. This gives us our separation from non-interactive assumptions.

2 Preliminaries

2.1 Digital Signature Scheme

We consider signature schemes that works over a set of common parameters, say GK. Concretely, there is a generator of the common parameters and the key generation algorithm takes GK as input. Such an extended formulation is often used in practical cryptographic protocols where many users share the group for efficiency reasons.

Definition 1 (Digital Signature Scheme). *A digital signature scheme* Sig *is a set of efficient algorithms* $(\mathcal{C}, \mathcal{K}, \mathcal{S}, \mathcal{V})$. $\mathcal{C}$ *is the common-parameter generator that takes security parameter* 1^λ *as input and outputs a common parameter* GK. $\mathcal{K}$ *is the key generator that takes* GK *as input and outputs a signing-key* SK *and verification-key* VK. *The keys include* GK *and the public-key defines a message space* Msp. $\mathcal{S}$ *is the signature generation algorithm that computes a signature* Σ *for input message* M *by using signing key* SK. $\mathcal{V}$ *is the verification algorithm that takes* VK, M, *and* Σ *and outputs* 1 *or* 0 *that represent acceptance and rejection, respectively.*

A signature scheme must be correct, i.e., it is required that for any keys generated by $\mathcal{K}$ and for any message in Msp, it holds that $1 = \mathcal{V}(VK, M, \mathcal{S}(SK, M))$. It is assumed that there exists an efficiently computable function $TstVk$ that takes λ and VK as input and checks the validity of VK such that if $0 \leftarrow TstVk(1^\lambda, VK)$ then $\mathcal{V}(VK, *, *)$ always returns 0, and if $1 \leftarrow TstVk(1^\lambda, VK)$ then the message space Msp is well defined and it is efficiently and uniformly sampleable. A signature Σ is called invalid (with respect to VK and M), if $1 \neq \mathcal{V}(VK, M, \Sigma)$. Otherwise, it is called valid.

We use the standard notion of existential unforgeability against adaptive chosen message attacks (EUF-CMA) [21] formally defined as follows.

Definition 2 (EUF-CMA). *A signature scheme* $\mathsf{Sig} = (\mathcal{C}, \mathcal{K}, \mathcal{S}, \mathcal{V})$ *is existentially unforgeable against adaptive chosen message attacks if, for any* $\mathcal{A} \in \mathsf{PPT}$, *the probability*

$$\Pr\left[\begin{array}{l} GK \leftarrow \mathcal{C}(1^\lambda), \\ (VK, SK) \leftarrow \mathcal{K}(GK), \\ (M^\star, \Sigma^\star) \leftarrow \mathcal{A}^{\mathcal{S}(SK,\cdot)}(VK) \end{array} : M^\star \notin Q \wedge 1 \leftarrow \mathcal{V}(VK, M^\star, \Sigma^\star)\right]$$

is negligible in λ. *Here,* $\mathcal{S}(SK, \cdot)$ *is a signing oracle that takes message* M *and returns signatures* $\Sigma \leftarrow \mathcal{S}(SK, M)$. Q *is the set of messages submitted to the signing oracle.*

2.2 Bilinear Groups

In this paper, let $\mathcal{G}$ be a generator of bilinear groups. It takes security parameter 1^λ as input and outputs $\Lambda := (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e)$ where

- p is a λ-bit prime,
- $\mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T$ are groups of prime order p with efficiently computable group operations, membership tests, and bilinear mapping $e : \mathbb{G}_1 \times \mathbb{G}_2 \to \mathbb{G}_T$,
- $\forall G \in \mathbb{G}_1 \setminus \{1\}, H \in \mathbb{G}_2 \setminus \{1\}$, $e(G, H)$ generates $\mathbb{G}_T$, and
- $\forall A \in \mathbb{G}_1, \forall B \in \mathbb{G}_2, \forall x, y \in \mathbb{Z}$: $e(A^x, B^y) = e(A, B)^{xy}$.

By generic operations, we mean the group operation, membership testing, and bilinear mapping over the groups in Λ. In Type-III groups [18], no efficient

isomorphisms are provided for either directions between $\mathbb{G}_1$ and $\mathbb{G}_2$. Throughout this paper, group descriptions Λ always describe Type-III groups.

By $\mathbb{G}_*$, we denote either $\mathbb{G}_1$ or $\mathbb{G}_2$ in Λ. For a vector of group elements $\boldsymbol{A} := (A_1, \ldots, A_k) \in \mathbb{G}_*^k$ and a vector of scalar values $\boldsymbol{x} := (x_1, \ldots, x_k) \in \mathbb{Z}_p^k$, we define the notation $\boldsymbol{A}^{\boldsymbol{x}} = \prod_{i=1}^k A_i^{x_i}$.

2.3 Structure Preserving Signatures

For a description of bilinear groups $\Lambda = (p, \mathbb{G}_1, \mathbb{G}_2, \mathbb{G}_T, e)$, an equation of the form

$$\prod_i \prod_j e(A_i, B_j)^{a_{ij}} = Z$$

for constants $a_{ij} \in \mathbb{Z}_p, Z \in \mathbb{G}_T$, and constants or variables $A_i \in \mathbb{G}_1$, $B_j \in \mathbb{G}_2$ is called a pairing product equation (PPE for short).

Definition 3 (Structure-Preserving Signatures). *A signature scheme* $(\mathcal{C}, \mathcal{K}, \mathcal{S}, \mathcal{V})$ *is called structure preserving with respect to bilinear group generator* $\mathcal{G}$ *if*

- *Common parameter GK consists of a group description* Λ*. Constants* a_{ij} *in* $\mathbb{Z}_p$ *are also included in GK if any,*
- *Verification-key VK includes* Λ *and group elements in* $\mathbb{G}_1$*,* $\mathbb{G}_2$*, and* $\mathbb{G}_T$*,*
- *Messages M consists of group elements in* $\mathbb{G}_1$ *and* $\mathbb{G}_2$*,*
- *Signature* Σ *consists of group elements in* $\mathbb{G}_1$ *and* $\mathbb{G}_2$*, and*
- *Verification* $\mathcal{V}$ *evaluates membership in* $\mathbb{G}_1$ *and* $\mathbb{G}_2$ *and PPEs.*

In a narrow sense, SPS might be limited to $Z = 1$ and *VK* excluding elements in $\mathbb{G}_T$ so that accompanying witness-indistinguishable Groth-Sahai proofs can have the zero-knowledge property.

2.4 Algebraic Algorithms

An algorithm is called algebraic with respect to a group if it takes a vector of elements $\boldsymbol{X}$ in the group and outputs a group element Y and there is a corresponding algorithm called an extractor that can output the representation of Y with respect to $\boldsymbol{X}$. For instance, if the algebraic algorithm $\mathcal{R}$ takes $A, B \in \mathbb{G}_*$ as input and outputs $C \in \mathbb{G}_*$, then $\mathcal{R}$'s extractor $\mathcal{E}$ outputs (a, b) such that $C = A^a B^b$.

In the following, we give a formal definition of the minimal case where an algorithm takes group elements from one group as input and outputs only one group element.

Definition 4 (Algebraic Algorithm). *Let* $\mathcal{R}$ *be a probabilistic polynomial time algorithm that takes* Λ*, a string* $aux \in \{0,1\}^*$*, and group elements* $\boldsymbol{X} \in \mathbb{G}_*^k$ *for some* k *and* $\mathbb{G}_*$ *in* Λ *as input and outputs a group element in* $\mathbb{G}_*$ *and a string* $ext \in \{0,1\}^*$*.* $\mathcal{R}$ *is called algebraic with respect to* $\mathcal{G}$ *if there exists* $\mathcal{E} \in \mathtt{PPT}$ *getting the same input as* $\mathcal{R}$ *including the same random coins such that for*

any $\Lambda \leftarrow \mathcal{G}(1^\lambda)$ *and all polynomial size* $\boldsymbol{X}$ *and* aux*, the following probability is negligible in* λ*.*

$$\Pr\left[\begin{matrix}(Y, ext) \leftarrow \mathcal{R}(\Lambda, \boldsymbol{X}, aux\,;\, r),\\ (\boldsymbol{y},\ ext) \leftarrow \mathcal{E}(\Lambda, \boldsymbol{X}, aux\,;\, r)\end{matrix} \ :\ Y \neq \boldsymbol{X}^{\boldsymbol{y}}\right].$$

Please note that unlike the case of the knowledge of exponent assumptions [12, 25, 6] that assumes the presence of $\mathcal{E}$ for *any malicious* $\mathcal{R}$, here we try to capture the limitation of current technology in building reduction algorithms. It is in fact easy to imagine an algorithm $\mathcal{R}$ that may not be algebraic as defined above; $\mathcal{R}$ takes a string from aux and directly translates it as a group element in $\mathbb{G}_*$. For such $\mathcal{R}$ there may not be an efficient extractor $\mathcal{E}$. However, a reduction algorithm that chooses Y in this way will typically not be more useful than one that chooses Y with a known discrete logarithm with respect to $\boldsymbol{X}$. Accordingly, we consider algorithms that compute on explicitly given group elements. We also stress that we are only interested in capturing the structure of Y with respect to the base $\boldsymbol{X}$. It is possible that aux contains additional group elements and that $\mathcal{R}$ returns group elements in ext for which we do not care to know a representation with respect to $\boldsymbol{X}$.

The above definition extends naturally to $\mathcal{A}$ that takes group elements from both groups and outputs multiple group elements at the same time. Furthermore, we note that algorithms that outputs no group elements can also be regarded as algebraic by taking the identity as default output for such algorithms so that extracting the representation is trivial. Trivial algorithms that output group elements taken from inputs intact are algebraic, too.

The notion is also extended to oracle algorithms. Let $(Y, ext)[\boldsymbol{X}', aux'] \leftarrow \bar{\mathcal{R}}^O(\Lambda, \boldsymbol{X}, aux)$ denote an execution of $\bar{\mathcal{R}}$ accessing to oracle O where $[\boldsymbol{X}', aux']$ denotes all inputs to $\bar{\mathcal{R}}$ given from (all invocations of) O. We say that oracle algorithm $\bar{\mathcal{R}}$ is algebraic if there exists an algebraic algorithm $\mathcal{R}$, and the computation by $\bar{\mathcal{R}}^O$ is equivalent to the following sequence of computation. First set $\boldsymbol{X}_0 := \boldsymbol{X}$ and $aux_0 := aux$. Run $(\boldsymbol{Y}_1, ext_1||\omega_1) \leftarrow \mathcal{R}(\Lambda, \boldsymbol{X}_0, aux_0)$ and repeat

$$\begin{aligned}&(\boldsymbol{X}'_i, aux'_i) \leftarrow O(\Lambda, \boldsymbol{Y}_i, ext_i),\\ &\boldsymbol{X}_{i+1} := \boldsymbol{X}_i||\boldsymbol{X}'_i, aux_{i+1} := \omega_i||aux'_i\\ &(\boldsymbol{Y}_{i+1}, ext_{i+1}||\omega_{i+1}) \leftarrow \mathcal{R}(\Lambda, \boldsymbol{X}_{i+1}, aux_{i+1}).\end{aligned}$$

for $i = 1$ until state ω_{i+1} explicitly indicates termination and $\boldsymbol{Y}_{i+1}$ includes Y. The extractor for $\bar{\mathcal{R}}$ is to compute $(\boldsymbol{y}, ext) \leftarrow \mathcal{E}^O(\Lambda, \boldsymbol{X}, aux)$ that fulfills $Y = (\boldsymbol{X}'')^{\boldsymbol{y}}$ for $\boldsymbol{X}'' = \boldsymbol{X} \cup \boldsymbol{X}'$. Such extractor can be constructed in straightforward manner by using the extractor for $\mathcal{R}$.

By $\mathtt{Cls}_{alb}$ we denote the set of all algebraic algorithms with respect to $\mathcal{G}$.

2.5 Non-interactive Hardness Assumptions

Intuitively, an assumption states that there is no algorithm $\mathcal{A}$ that is better than any known (typically trivial) algorithm U, which, for example, selects its

output uniformly from a proper domain. In fact, our formulation is so general that it can capture too strong assumptions that never hold and too weak ones that always hold. But it does not matter for our purpose since we are to show the impossibility to reduce the security of a signature scheme to such (extreme) assumptions.

Definition 5 (Non-interactive Hardness Assumptions). *A non-interactive problem consists of a triple of algorithms $P = (I, V, U)$ where $I \in \mathtt{PPT}$ is an instance generator, which takes 1^λ and outputs a pair of an instance and a witness, (y, w), and V is a verification algorithm that takes y, w and an answer x, and outputs 1 or 0 that represents acceptance or rejection, respectively. A non-interactive hardness assumption for problem P is to assume that, for any $\mathcal{A} \in \mathtt{PPT}$, the following advantage function Adv is negligible in λ.*

$$\begin{aligned} Adv_{\mathcal{A}}(1^\lambda) = &\Pr[(y,w) \leftarrow I(1^\lambda), x \leftarrow \mathcal{A}(y) \,:\, 1 = V(y,x,w)] \\ &- \Pr[(y,w) \leftarrow I(1^\lambda), x \leftarrow U(y) \,:\, 1 = V(y,x,w)] \end{aligned} \tag{1}$$

In search problems, U is typically set to an algorithm that returns constant $\perp$ (or a random answer x when the domain is uniformly sampleable). In decision problems, U typically returns 1 or 0 randomly so that the latter probability is $1/2$.

As we are concerned with structure preserving signatures, we consider hard problems that are defined over bilinear groups as follows.

Definition 6 (Hard Problem over $\mathcal{G}$). *A non-interactive problem P over bilinear group generator $\mathcal{G}$ is a non-interactive problem such that*

- *instance generator I runs $\Lambda \leftarrow \mathcal{G}(1^\lambda)$, and output y includes Λ, and*
- *there exists $\mathcal{A}$ that solves P with access to an oracle that solves the discrete logarithm problem for the groups in Λ.*

By $\mathtt{NIP}$, we denote all non-interactive problems. Similarly, $\mathtt{NIP}_{\mathcal{G}}$ denotes $\mathtt{NIP}$ over $\mathcal{G}$. Throughout the paper, we simply say that algorithm $\mathcal{A}$ solves problem P if advantage $\mathrm{Adv}_{\mathcal{A}}(1^\lambda)$ is not negligible.

2.6 Black-Box Reduction and Meta-Reduction

When algorithm $\mathcal{R}$ is given $\mathcal{A}$ as black-box, denoted by $\mathcal{R}^{\mathcal{A}}$, we mean that $\mathcal{R}$ and $\mathcal{A}$ are given the same security parameter and $\mathcal{A}$ is given access to arbitrary number of copies of $\mathcal{A}$ as oracles. Interaction between $\mathcal{R}$ and $\mathcal{A}$ can be done in interleaving manner. If $\mathcal{A}$ is a randomized algorithm, $\mathcal{A}$ has random coins inside and every copy uses the same randomness. The security parameter and the random coins are out of the control of $\mathcal{R}$.

For problem P and signature scheme Sig, $\mathcal{R}$ is a fully black-box reduction if, for any (even inefficient) successful forger $\mathcal{A}$ for Sig, $\mathcal{R}^{\mathcal{A}}$ is successful in solving P. By $\mathsf{Sig} \Rightarrow_{\mathcal{R}} P$, we mean that R is a black-box reduction from Sig to P. A separation between Sig and P is to show that for Sig and P, there is no such $\mathcal{R}$

under hardness assumption for problem P'. (The problem P' can be the same as P to make the separation unconditional.) Note that $\mathcal{R}$ depends on Sig and P. To claim that a class of hardness assumption falls short of proving the security of any construction of a signature scheme in a class by any black-box reduction, one need to show the absence of $\mathcal{R}$ for every signature and assumption in the respective classes.

In the meta-reduction paradigm, a proof typically begin with constructing a magic adversary $\mathcal{A}$ that is inefficient (or given access to powerful oracle) but successful in breaking Sig so that $\mathcal{R}^{\mathcal{A}}$ works as expected. It then constructs meta-reduction $\mathcal{M}$ that $\mathcal{M}^{\mathcal{R}}$ solves P'. A major task of $\mathcal{M}$ is to efficiently emulate $\mathcal{A}$ by rewinding $\mathcal{R}$ and/or exploiting special properties of $\mathcal{R}$ and Sig. If $\mathcal{M}$ is successful in the emulation, $\mathcal{M}^{\mathcal{R}}$ can be seen as a polynomial-time algorithm that solves P', which contradicts the assumed hardness of P'.

3 Crucial Relation

If any algorithm that simulates signatures must "know" the secret key, the unforgeability of the signature scheme cannot be proven by black-box reduction to any non-interactive assumption. We extend this idea in such a way that it is not necessary to know the entire secret key but some *crucial information* is necessary to conduct the simulation and sufficient to forge a signature if leaked to the adversary. Informally, crucial information is a witness for a binary relation, $\Psi(\theta, \varpi)$, which we call *crucial relation* defined over signatures θ and some sensitive information ϖ. The relation requires three properties: every θ has exactly one ϖ (uniqueness), whenever an entity is successful in producing signatures, it is possible to extract ϖ from the entity (extractability), and ϖ is useful enough to yield a forgery (usefulness). A crucial relation is defined with respect to a class of algorithms, $\mathtt{Cls} \subseteq \mathtt{PPT}$ to which the entity that generates θ belongs.

Let us first prepare some notations used in the formal definition. For a public key VK, a sequence of messages $\boldsymbol{M} = \{M_1, \ldots, M_n\} \in \mathsf{Msp}^n$ and signatures $\boldsymbol{\Sigma} = \{\Sigma_1, \ldots, \Sigma_n\}$, define $\mathcal{V}(\theta)$ for $\theta := (VK, \boldsymbol{M}, \boldsymbol{\Sigma})$ by a function that returns $\prod_{i=1}^{n} \mathcal{V}(VK, M_i, \Sigma_i)$.

Definition 7 (Crucial Relation). *Let $\mathsf{Sig} = (\mathcal{C}, \mathcal{K}, \mathcal{S}, \mathcal{V})$ be a signature scheme. Let $\varpi \in \{0,1\}^*$ and $\theta = (VK, \boldsymbol{M}, \boldsymbol{\Sigma}) \in \{0,1\}^*$. A relation $\Psi(\theta, \varpi)$ is a crucial relation for Sig with respect to a class of algorithms $\mathtt{Cls}$ if the following properties are provided.*

- *(Uniqueness) For every $\theta := (VK, \boldsymbol{M}, \boldsymbol{\Sigma})$ such that $1 = \mathcal{V}(\theta)$, there exists exactly one (polynomial size) ϖ fulfilling $1 = \Psi(\theta, \varpi)$.*
- *(Extractability) For any $\mathcal{R} \in \mathtt{Cls}$, there exists $\mathcal{E} \in \mathtt{PPT}$ and $n > 0$ such that, for any $VK \in \{0,1\}^*$ such that $1 \leftarrow TstVk(1^\lambda, VK)$, and any arbitrary string φ in $1^\lambda || \{0,1\}^*$, probability*

$$\Pr\left[\begin{array}{l} \boldsymbol{M} \leftarrow \mathsf{Msp}^n \\ \boldsymbol{\Sigma} \leftarrow \mathcal{R}(\varphi, \boldsymbol{M}) \\ \varpi \leftarrow \mathcal{E}(\varphi, \boldsymbol{M}) \\ \theta := (VK, \boldsymbol{M}, \boldsymbol{\Sigma}) \end{array} : 1 = \mathcal{V}(\theta) \wedge 1 \neq \Psi(\theta, \varpi)\right] \tag{2}$$

is negligible in λ. The probability is taken over the choice of $\boldsymbol{M}$ and the randomness given to $\mathcal{R}$. The same randomness is given to $\mathcal{E}$.

– *(Usefulness) There exists an algorithm $\mathcal{B} \in \mathtt{PPT}$ such that, for any $\theta := (VK, \boldsymbol{M}, \boldsymbol{\Sigma})$ and ϖ that satisfies $\Psi(\theta, \varpi) = 1$, the following probability is not negligible in λ.*

$$\Pr\left[(M, \Sigma) \leftarrow \mathcal{B}(\theta, \varpi) \ : \ M \notin \boldsymbol{M} \wedge 1 = \mathcal{V}(VK, M, \Sigma)\right]$$

Remarks:

- The intuition of extractability is that whenever φ is helpful for $\mathcal{R}$ in computing valid signatures, extractor $\mathcal{E}$ should be successful in extracting ϖ from φ. This must hold even for non-legitimate VK as long as it is functional with respect to the verification.
- For $\mathcal{R}$ that is successful only with negligible probability, $\mathcal{E}$ can be an empty algorithm. So we only need to care for successful $\mathcal{R}$ that yields valid signatures. In particular, conditioned that $1 = \mathcal{V}(\theta)$ happens with noticeable probability, the conditional provability that $1 = \Psi(\theta, \varpi)$ is overwhelming.
- There may be many φ that make $\mathcal{R}$ produce the same $\boldsymbol{\Sigma}$ from the same VK and $\boldsymbol{M}$. Whichever φ is given, $\mathcal{E}$ must output the same ϖ.

Let $\mathtt{SIGCR}_{\mathtt{Cls}}$ denote signature schemes that has a crucial relation for a class of algorithms, $\mathtt{Cls}$. We require $\mathtt{Cls}$ be a class of algorithms in $\mathtt{PPT}$ that satisfies the following trivial composition. For any $\mathcal{A} \in \mathtt{Cls}$, the following $\mathcal{A}'$ is also in $\mathtt{Cls}$. $\mathcal{A}'$ takes inputs, say aux_1 and $X_1, \ldots, X_n$, and runs $\mathcal{A}$ as $(aux_{i+1}, Y_{i+1}) \leftarrow \mathcal{A}(aux_i, X_i)$ for $i = 1, ..., n$. $\mathcal{A}$ then picks some Y_i whose index is in the list specified in aux_1. Obviously, algebraic algorithms are in such a class. The following proof is given for such $\mathtt{Cls}$.

Theorem 8. *For any signature scheme Sig in $\mathtt{SIGCR}_{\mathtt{Cls}}$, for any non-interactive problem P in $\mathtt{NIP}$, there is no $\mathcal{R} \in \mathtt{Cls}$ such that $\mathsf{Sig} \Rightarrow_{\mathcal{R}} P$ if pseudo-random functions exit.*

Proof. Let O be a deterministic oracle that takes θ as input and returns ϖ that $1 = \Psi(\theta, \varpi)$ if it exists (otherwise return $\perp$). Consider the following all-powerful adversary $\mathcal{A}$ attacking Sig with access to O. Let f be a pseudo-random function. Given VK as input, $\mathcal{A}$ selects a random key for f and checks if $1 \leftarrow TstVk(1^\lambda, VK)$ (if not, $\mathcal{A}$ halts). Then it chooses $\boldsymbol{M}$ randomly from Msp^n for some constant n by using pseudo-randomness generated by $f(VK)$. Let $\boldsymbol{M} \leftarrow \mathsf{Msp}^f(VK)$ denote these steps. $\mathcal{A}$ then send $\boldsymbol{M}$ to the signing oracle (simulated by $\mathcal{R}$). After receiving n signatures, $\boldsymbol{\Sigma}$, $\mathcal{A}$ aborts if $\boldsymbol{\Sigma}$ contains an invalid signature. Otherwise, $\mathcal{A}$ calls O with input $\theta = (VK, \boldsymbol{M}, \boldsymbol{\Sigma})$ and obtains ϖ. It then executes $(M, \Sigma) \leftarrow \mathcal{B}(VK, \boldsymbol{M}, \boldsymbol{\Sigma}, \varpi)$ and outputs (M, Σ).

To verify that above $\mathcal{A}^O$ is indeed a successful forger, consider that $\mathcal{A}^O$ is given legitimate VK and signatures generated by $\mathcal{S}(SK, \boldsymbol{M})$. By correctness of Sig and the uniqueness property, ϖ indeed exist and is uniquely defined. So O returns ϖ. Then due to the usefulness property, the output from $\mathcal{B}$ satisfies the

predicates with probability not negligible in λ. Thus $\mathcal{A}^O$ is a successful forger against Sig.

Suppose that there exists $\mathcal{R} \in \mathtt{Cls}$ that $\mathsf{Sig} \Rightarrow_{\mathcal{R}} P$ holds. Since $\mathcal{R}$ is a fully black-box reduction it must be successful with the above $\mathcal{A}^O$. Namely, $\mathrm{Adv}_{\mathcal{R}^{\mathcal{A}^O}}(1^\lambda)$ as defined in Definition 5 is not negligible.

Without loss of generality, we assume that $\mathcal{A}$ outputs n messages as $\boldsymbol{M}$ at once. We also assume, without loss of generality, that when $\mathcal{R}$ outputs something for interaction it also outputs the internal state φ at that moment. Then $\mathcal{R}$ is restarted taking φ and some data from the interaction as input.

We construct meta-reduction $\mathcal{M}$ that $\mathcal{M}^{\mathcal{R}}$ solves P. $\mathcal{M}$ emulates $\mathcal{A}^O$ without any oracles. By a session, we mean the conversation between $\mathcal{R}$ and a copy of $\mathcal{A}$ initiated by $\mathcal{R}$ with input VK_i to $\mathcal{A}$. Every session is labelled by an index. Given $y \leftarrow I(1^\lambda)$, $\mathcal{M}$ sets $\varphi_0 := y$. Let $\mathsf{BADSIG}[i]$ be a flag that indicates the presence of an invalid signature in i-th session. It is initialized to zero. $\mathcal{M}$ runs $\mathcal{R}(\varphi_0)$ and do as follows.

- If $\mathcal{R}$ outputs (φ_i, VK_j) to invoke j-th copy of $\mathcal{A}$, $\mathcal{M}$ checks $TstVk(1^\lambda, VK_j)$ and halt the session if it is not 1. Otherwise, $\mathcal{M}$ selects $\boldsymbol{M}_j \leftarrow \mathsf{Msp}_j^n$ (if the same VK_j has been observed before, say in session k, $\mathcal{M}$ uses the same $\boldsymbol{M}_k$ instead), and resume $\mathcal{R}$ as $\mathcal{R}(\varphi_i || \boldsymbol{M}_j)$. Here Msp_j is the message space associated to VK_j.
- If $\mathcal{R}$ outputs $(\varphi_i, \Sigma_{k,\ell})$ for existing session k, $\mathcal{M}$ checks if $1 = \mathcal{V}(VK_k, M_{k,\ell}, \boldsymbol{\Sigma}_{k,\ell})$. If not, $\mathcal{M}$ sets $\mathsf{BADSIG}[k]$ to 1. It then continues as follows.
  - If $\ell < n$, $\mathcal{M}$ continues by running $\mathcal{R}(\varphi_i)$.
  - If $\ell = n$ and $\mathsf{BADSIG}[k] = 0$, then $\mathcal{M}$ extracts ϖ_k for this session as follows. Let φ_i be the internal state that $\mathcal{R}$ outputs with VK_k. Let $\boldsymbol{M}_{k'}$ be the last message $\mathcal{R}$ is given before outputting $\Sigma_{k,n}$. Let $\varphi'_i := \varphi_i || \{\boldsymbol{M}_{k+1}, \ldots, \boldsymbol{M}_{k'}\}$. Let $\mathcal{R}'$ be an algorithm associated to $\mathcal{R}$ that computes $\boldsymbol{\Sigma}_k \leftarrow \mathcal{R}'(\varphi'_i, \boldsymbol{M}_k)$. $\mathcal{R}'$ is a simple algorithm that parses φ'_i into $\varphi_i || \{\boldsymbol{M}_{k+1}, \ldots, \boldsymbol{M}_{k'}\}$, runs $\mathcal{R}(\varphi_i, \boldsymbol{M}_k)$, continue running $\mathcal{R}$ giving messages $\boldsymbol{M}_{k+1}, \ldots, \boldsymbol{M}_{k'}$ as input, and collects signatures $\Sigma_{k,i}$ for $i = 1, \ldots, n$, and finally outputs $\boldsymbol{\Sigma}_k$. As $\mathcal{R}$ is in $\mathtt{Cls}$, so is $\mathcal{R}'$ as assumed to $\mathtt{Cls}$. Due to the extractability property, there exists polynomial-time $\mathcal{E}$ that computes ϖ_k for $\theta_k := (VK_k, \boldsymbol{M}_k, \boldsymbol{\Sigma}_k)$. Thus, $\mathcal{M}$ runs $\mathcal{E}(\varphi'_i, \boldsymbol{M}_k)$ and obtains ϖ_k. As $\mathcal{V}(\theta_k) = 1$ holds, $1 = \Psi(\theta_k, \varpi_k)$ holds except for negligible probability. $\mathcal{M}$ then invokes $(M^\star{}_k, \Sigma^\star{}_k) \leftarrow \mathcal{B}(\theta_k, \varpi_k)$ and runs $\mathcal{R}(\varphi_i || (M^\star{}_k, \Sigma^\star{}_k))$ to continue.
- If $\mathcal{R}$ outputs x, then $\mathcal{M}$ outputs x and halts.

Let $\mathrm{Adv}^P_{\mathcal{M}^{\mathcal{R}}}(1^\lambda)$ be the advantage of the above $\mathcal{M}$ in solving P. We show that the difference $|\mathrm{Adv}^P_{\mathcal{R}^{\mathcal{A}^O}}(1^\lambda) - \mathrm{Adv}^P_{\mathcal{M}^{\mathcal{R}}}(1^\lambda)|$ is negligible. We start from $\mathcal{M}^{\mathcal{R}}$ and modifies $\mathcal{M}$ slightly at a time. First replace truly random choice $\boldsymbol{M}_j \leftarrow \mathsf{Msp}_j^n$ with pseudo-random one $\boldsymbol{M} \leftarrow \mathsf{Msp}^f(VK)$. Call this modified algorithm $\mathcal{M}'$. The loss of the advantage by this modification is negligible due to the indistinguishability of f. We prove that by constructing a distinguisher $\mathcal{D}$ for f as follows. $\mathcal{D}$ runs $(y, w) \leftarrow I(1^\lambda)$ and emulate $\mathcal{M}^{\mathcal{R}}(y)$ as it is except that whenever $\mathcal{M}$ chooses

$\boldsymbol{M}_k$, $\mathcal{D}$ sends VK_k to the challenger and obtains a string and use it as random coins to generate $\boldsymbol{M}_k$. It then returns $\boldsymbol{M}_k$ to $\mathcal{R}$. When $\mathcal{M}$ terminates with x, $\mathcal{D}$ outputs $V(y, x, w)$. Obviously, if the strings from the challenger are truly random, $\mathcal{D}$ emulates $\mathcal{M}$. If, on the other hand, they are the output of f, $\mathcal{D}$ emulates $\mathcal{M}'$. Since the advantage of $\mathcal{D}$, say $\mathrm{Adv}^f_{\mathcal{D}}(1^\lambda)$, is assumed negligible, we have $|\mathrm{Adv}^P_{\mathcal{M}}(1^\lambda) - \mathrm{Adv}^P_{\mathcal{M}'}(1^\lambda)| = \mathrm{Adv}^f_{\mathcal{D}}(1^\lambda) < \mathrm{negl}(\lambda)$.

Next replace extractor $\mathcal{E}$ with oracle O. Call this modified algorithm $\mathcal{M}''$. We show that the loss of advantage by moving from $\mathcal{M}'$ to $\mathcal{M}''$ is negligible. Let

$$\Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^n_j \\ \varpi \leftarrow \mathcal{E}\end{matrix}\right] \tag{3}$$

denote the probability presented in (2). We replace Msp^n_j and $\mathcal{E}$ with Msp^f and O accordingly with trivial meaning. With this notation, the loss of advantage is upper bound by

$$|\mathrm{Adv}^P_{\mathcal{M}'}(1^\lambda) - \mathrm{Adv}^P_{\mathcal{M}''}(1^\lambda)| \leq \left|\Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^f \\ \varpi \leftarrow \mathcal{E}\end{matrix}\right] - \Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^f \\ \varpi \leftarrow O\end{matrix}\right]\right|. \tag{4}$$

To evaluate the right hand of (4), first observe that

$$\left|\Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^n_j \\ \varpi \leftarrow \mathcal{E}\end{matrix}\right] - \Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^f \\ \varpi \leftarrow \mathcal{E}\end{matrix}\right]\right| \tag{5}$$

is negligible due to the indistinguishability of f. Also,

$$\left|\Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^n_j \\ \varpi \leftarrow \mathcal{E}\end{matrix}\right] - \Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^n_j \\ \varpi \leftarrow O\end{matrix}\right]\right| \tag{6}$$

is negligible due to the extractability property. Finally observe that

$$\left|\Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^n_j \\ \varpi \leftarrow O\end{matrix}\right] - \Pr\left[\begin{matrix}\boldsymbol{M} \leftarrow \mathsf{Msp}^f \\ \varpi \leftarrow O\end{matrix}\right]\right| \tag{7}$$

is zero because oracle O never causes $1 \neq \Psi(\theta, \varpi)$ if $1 = \mathcal{V}(\theta)$ due to the uniqueness condition. Thus both probabilities in (7) are zero. Since (5) to (7) are all negligible, we conclude that (4) is negligible, too.

Finally, observe that $\mathcal{M}''$ is identical to $\mathcal{A}^O$. Accordingly, $|\mathrm{Adv}^P_{\mathcal{R},\mathcal{A}^O}(1^\lambda) - \mathrm{Adv}^P_{\mathcal{M}^{\mathcal{R}}}(1^\lambda)|$ is negligible. Since $\mathcal{R}$ and $\mathcal{E}$ belongs to $\mathtt{Cls} \subseteq \mathtt{PPT}$ and $\mathcal{M}$ only performs operations that can be done in polynomial-time, the total running time of $\mathcal{M}$ and $\mathcal{R}$ remains polynomial. Thus $\mathcal{M}^{\mathcal{R}}$ forms a polynomial-time algorithm that solves P, which contradicts to the assumed hardness of P. □

4 Crucial Relation in Size-3 SPS

We consider the class of algebraic reductions that make oracle calls with keys formed over over the groups for which it is defined as algebraic. This constraint

plays a role when we construct an extractor for crucial relation based on the extractor associated with the algebraic reduction. Since the extractor works only for the groups the algebraic reduction is defined, so does the resulting extractor for crucial relation. Since the crucial relation involves the verification keys, we require all keys to be generated over the same groups the extractor works for. We call such algorithms *group-preserving algebraic reductions*. This notion has been used before in the literature, e.g., [19] and the constraint also has some similarity to key-preservation [31] and instance non-malleability [29].

Theorem 9. *There exists no group-preserving algebraic reduction that reduces the existential unforgeability of an SPS scheme to hardness of any problem in* $\mathtt{NIP}_{\mathcal{G}}$ *if signatures consist of three base group elements.*

We prove Theorem 9 actually by proving the following lemma. Then applying Theorem 8 completes the proof.

Lemma 10. *Any SPS scheme with signature size 3 has a crucial relation with respect to group-preserving algebraic algorithms.*

We begin by recalling the result from [3] that any SPS scheme whose verification consists of one pairing product equation, or whose signature consists only of $\mathbb{G}_1$ or $\mathbb{G}_2$ is not EUF-CMA. A signature scheme for signing multiple elements at once can always be used to sign a single element by setting the other group elements to 1. Without loss of generality, it therefore suffices to consider schemes whose message consists of a single group element and where the signature consists of 2 elements in one group and 1 element in the other. We will also consider, without loss of generality, the case where the verification consists of two pairing product equations. The result applies to schemes with more than two verification equations as well and the proofs can be adopted with superficial changes.

Case of $\Sigma \in \mathbb{G}_1^2 \times \mathbb{G}_2$.
In any SPS whose signature consists of 3 group elements, $(R, S, T) \in \mathbb{G}_1^2 \times \mathbb{G}_2$, the verification predicate includes at least two pairing product equations that can be reduced to the following general form.

$$e(R, U_1\, T^{a_1})\, e(S, U_2\, T^{a_2})\, e(M, U_3\, T^{a_3})\, e(U_0, T^{a_4}) = Z_1 \tag{8}$$

$$e(R, V_1\, T^{b_1})\, e(S, V_2\, T^{b_2})\, e(M, V_3\, T^{b_3})\, e(V_0, T^{b_4}) = Z_2 \tag{9}$$

The group elements except for M, R, S and T are taken from the public key, and the constants in $\mathbb{Z}_p$ are taken from the common parameters. For a message M and a signature (R, S, T), let φ_r, α_r, φ_s, α_s, and t be

$$R = G^{\varphi_r} M^{\alpha_r}, \quad S = G^{\varphi_s} M^{\alpha_s}, \quad \text{and} \quad T = H^t. \tag{10}$$

We consider φ_r, α_r, φ_s, α_s be variables that fulfill relations determined by (8), (9) and (10). Let f_1 and f_2 be

$$f_1 = \alpha_r m + \varphi_r - r, \quad \text{and} \quad f_2 = \alpha_s m + \varphi_s - s \tag{11}$$

where small-case letters, r, s, and m, represents the discrete-logs (to base G) of group elements denoted by corresponding large-case letters. (This convention is used throughout this paper.) By replacing R and S in (8) with those in (10) and taking the discrete-logs with respect to base $e(G, H)$, we can represent (8) as $f_3 m + f_4 = 0$ where

$$f_3 = \alpha_r\,(u_1 + a_1\,t) + \alpha_s\,(u_2 + a_2\,t) + (u_3 + a_3\,t), \text{ and} \tag{12}$$

$$f_4 = \varphi_r\,(u_1 + a_1\,t) + \varphi_s\,(u_2 + a_2\,t) + u_0\,a_4\,t - z_1. \tag{13}$$

Similarly, (9) can be represented as $f_5 m + f_6 = 0$ where

$$f_5 = \alpha_r\,(v_1 + b_1\,t) + \alpha_s\,(v_2 + b_2\,t) + (v_3 + b_3\,t), \text{ and} \tag{14}$$

$$f_6 = \varphi_r\,(v_1 + b_1\,t) + \varphi_s\,(v_2 + b_2\,t) + v_0\,b_4\,t - z_2. \tag{15}$$

Consider a system of equations $Q := \{f_1 = 0, \ldots, f_6 = 0\}$. Focus on a non-redundant part, e.g., $f_1 = f_2 = f_3 = f_5 = 0$ which is represented as

$$\begin{pmatrix} m & 0 & 1\ 0 \\ 0 & m & 0\ 1 \\ u_1 + a_1 t & u_2 + a_2 t & 0\ 0 \\ v_1 + b_1 t & v_2 + b_2 t & 0\ 0 \end{pmatrix} \cdot \begin{pmatrix} \alpha_r \\ \alpha_s \\ \varphi_r \\ \varphi_s \end{pmatrix} = \begin{pmatrix} r \\ s \\ -(u_3 + a_3 t) \\ -(v_3 + b_3 t) \end{pmatrix}. \tag{16}$$

Let K_t denote the leftmost matrix in (16). It has rank 4, and

$$\det(K_t) = (a_1 b_2 - a_2 b_1)\,t^2 + (a_1 v_2 + u_1 b_2 - u_2 b_1 - a_2 v_1)\,t + (u_1 v_2 - u_2 v_1). \tag{17}$$

If $\det(K_t) \neq 0$, there exists unique $(\alpha_r, \alpha_s, \varphi_r, \varphi_s)$ that fulfills Q. Note that Q is defined with respect to the public key and M and T.

Crucial Relation. Now we are ready to define a crucial relation as follows. For $VK = (GK, U_0, U_1, U_2, U_3, U_4, V_0, V_1, V_2, V_3, V_4)$ and $\theta = (VK, \boldsymbol{M}, \boldsymbol{\Sigma})$, let $\varpi = (\alpha_r, \alpha_s, G^{\varphi_r}, G^{\varphi_s}, H^t)$. Relation $\Psi(\theta, \varpi)$ returns 1 if there exists a valid (M, R, S, T) in θ such that

- $T = H^t$,
- $(\alpha_r, \alpha_s, \varphi_r, \varphi_s)$ determined by ϖ fulfills Q w.r.t. VK and M, and
- (M, R, S, T) is the first one in θ that $\det(K_t) \neq 0$.

Relation Ψ also returns 1 if $\det(K_t) = 0$ for all (M, R, S, T) in θ and $\varpi = \perp$. Note that the second condition implies $R = G^{\varphi_r} M^{\alpha_r}$, $S = G^{\varphi_s} M^{\alpha_s}$. Such ϖ is extractable, unique, and useful as shown below.

Uniqueness. The first (M, Σ) with $\det(K_t) \neq 0$ is unique in θ (assuming that signatures are stored in order) if it exists. Then, ϖ is uniquely determined for such (M, Σ) from relation (16). When there is no (M, σ) with $\det(K_t) \neq 0$ exists in θ, ϖ is also uniquely defined to $\perp$. Accordingly, for any θ, there is unique ϖ such that $\Psi(\theta, \varpi) = 1$.

USEFULNESS. Given ϖ that satisfies $\Psi(\theta, \varpi) = 1$, a valid signature for arbitrary message can be created as follows. We first consider the case where $\varpi = (\alpha_r, \alpha_s, G^{\varphi_r}, G^{\varphi_s}, H^t) \neq \perp$. Given ϖ and arbitrary message $M^\star$, compute $R^\star = (G^{\varphi_r})M^{\star\alpha_r}$, $S^\star = (G^{\varphi_s})M^{\star\alpha_s}$, $T^\star = (H^t)$. To see that $\Sigma^\star = (R^\star, S^\star, T^\star)$ is a valid signature for $M^\star$, observe that the first verification predicate (8) is

$$\begin{aligned}
&e(R^\star, U_1 T^{a_1})\, e(S^\star, U_2 T^{a_2})\, e(M^\star, U_3 T^{a_3})\, e(U_0, U_4 T^{a_4}) \\
&= e(G^{\varphi_r} M^{\star\alpha_r}, H^{u_1 + a_1 t})\, e(G^{\varphi_s} M^{\star\alpha_s}, H^{u_2 + a_2 t}) \\
&\qquad e(M^\star, H^{u_3 + a_3 t})\, e(G^{u_0}, H^{u_4 + a_4 t}) \\
&= e(M^\star, H)^{f_3}\, e(G, H)^{f_4}.
\end{aligned}$$

It results in 1 since ϖ satisfies $f_3 = f_4 = 0$. The second predicate can be verified in the same way. Thus, by choosing fresh $M^\star$, $(R^\star, S^\star, T^\star)$ is a successful forgery.

We next consider the case of $\varpi = \perp$. It means that $\det(K_t) = 0$ holds for all M and (R, S, T) in θ. We then present a concrete attack as follows. First we consider the case where (17) is not a zero polynomial. Since (17) is quadratic in t, there are at most two Ts for which $\det(K_t) = 0$. Given θ including more than three signatures, such T must appear more than once. Given two signatures (M_1, R_1, S_1, T) and (M_2, R_2, S_2, T) in θ, the forger computes random linear combination of the signatures as $(M^\star, R^\star, S^\star) = (M_1^{\beta_1} M_2^{\beta_2}, R_1^{\beta_1} R_2^{\beta_2}, S_1^{\beta_1} S_2^{\beta_2})$ for randomly chosen β_1 and β_2 that satisfies $\beta_1 + \beta_2 = 1$. Then $(R^\star, S^\star, T)$ is a valid signature for $M^\star$ that is random and fresh with high probability. (The forger chooses messages that are not 1 to make sure $M_1 \neq 1$ or $M_2 \neq 1$ to get $M^\star$ uniform.) Next consider the case where (17) is a zero polynomial. Then we have $a_1 b_2 = a_2 b_1$ and $u_1 v_2 = u_2 v_1$. Let δ_1 and δ_2 be

$$\delta_1 := \frac{b_1}{a_1} = \frac{b_2}{a_2}, \quad \text{and} \quad \delta_2 := \frac{v_1}{u_1} = \frac{v_2}{u_2}, \tag{18}$$

which are defined to zero if any of a_1, a_2, u_1 or u_2 is zero. Then, from $f_3 = f_5 = 0$ in (12) and (14), we have

$$\left(\frac{u_2 + a_2 t}{u_1 + a_1 t} - \frac{v_2 + b_2 t}{v_1 + b_1 t}\right) \alpha_s + \left(\frac{u_3 + a_3 t}{u_1 + a_1 t} - \frac{v_3 + b_3 t}{v_1 + b_1 t}\right) = 0. \tag{19}$$

The coefficient of α_s in (19) is zero since $\det(K_t) = 0$. Thus we have

$$\frac{u_3 + a_3 t}{u_1 + a_1 t} - \frac{v_3 + b_3 t}{v_1 + b_1 t} = 0. \tag{20}$$

Since (20) holds for any t, we have

$$\frac{b_3}{a_3} = \frac{b_1}{a_1} = \delta_1, \quad \text{and} \quad \frac{v_3}{u_3} = \frac{v_1}{u_1} = \delta_2. \tag{21}$$

Similarly, from $f_4 = f_6 = 0$ in (13) and (15), we have

$$\frac{v_0\, b_4}{u_0\, a_4} = \frac{b_1}{a_1} = \delta_1, \quad \text{and} \quad \frac{z_2}{z_1} = \frac{v_1}{u_1} = \delta_2. \tag{22}$$

From (18), (21) and (22), the second verification predicate (9) is

$$1 = e(R^{a_1} S^{a_2} M^{a_3} U_0^{a_4}, T)^{\delta_1} \cdot \{e(R, U_1)\, e(S, U_2)\, e(M, U_3)\, Z_1^{-1}\}^{\delta_2},$$

and the first verification predicate (8) is

$$1 = e(R^{a_1} S^{a_2} M^{a_3} U_0^{a_4}, T) \cdot \{e(R, U_1)\, e(S, U_2)\, e(M, U_3)\, Z_1^{-1}\}.$$

If $\delta_1 = \delta_2$, the verification predicates are in a linear relation. Thus they shrink into one predicate and the scheme is insecure. If $\delta_1 \neq \delta_2$, the equations hold if and only if

$$e(R^{a_1} S^{a_2} M^{a_3} U_0^{a_4}, T) = 1, \quad \text{and} \quad e(R, U_1)\, e(S, U_2)\, e(M, U_3)\, Z_1^{-1} = 1.$$

The first equation implies either $R^{a_1} S^{a_2} M^{a_3} U_0^{a_4} = 1$ or $T = 1$. For such a case, the following attack succeeds. Request three or more signatures on randomly chosen messages. Then find two signatures (M_1, R_1, S_1, T_1) and (M_2, R_2, S_2, T_2) such that $T_1 = T_2 = 1$ or $T_1 \cdot T_2 \neq 1$. Then, linear combination of the two signatures yields a new valid signature. That is, let $(M^\star, R^\star, S^\star) = (M_1^{\beta_1} M_2^{\beta_2}, R_1^{\beta_1} R_2^{\beta_2}, S_1^{\beta_1} S_2^{\beta_2})$ for randomly chosen β_1 and β_2 that satisfies $\beta_1 + \beta_2 = 1$. Then $(M^\star, R^\star, S^\star, T_1)$ is a valid fresh signature. Keeping the condition on T_1 and T_2 in mind, inspection is not hard and omitted. This concludes that a successful forgery is possible even for the case of $\varpi = \perp$.

Extractability. Observe that, for any algebraic algorithm that obtains M as input and computes group element R, there exists an extractor that outputs α_r such that $R = (G^{\varphi_r}) M^{\alpha_r}$ where (G^{φ_r}) part is computed by multi-base exponentiation of group elements except for M. Similarly, the extractor outputs α_s such that $S = (G^{\varphi_s}) M^{\alpha_s}$. Thus $(\alpha_1, \alpha_2, \varphi_1, \varphi_2)$ determined uniquely from extracted $(\alpha_1, \alpha_2, G^{\varphi_1}, G^{\varphi_2}, H^t)$ fulfills f_1 and f_2. We then claim that $f_i = 0$ for $i = 3, \ldots, 6$ also hold except for negligible probability. Otherwise, the algorithm can be used to solve the discrete-logarithm problem between G and M. As we can manipulate all group elements given to the algorithm so that all their discrete-logarithms are known except for M, we can compute φ_r (and φ_s) from the extracted exponents. Suppose that, without loss of generality, $f_3 \neq 0$ happens for $M \neq 1$. Since $f_3 m + f_4 = 0$ for valid signature, $f_4 \neq 0$ happens, too. Thus equation $f_3 m + f_4 = 0$ with non-zero f_3 and f_4 determine m. For the case of $f_5 \neq 0$, use equation $f_5 m + f_6 = 0$ with non-zero f_5 and f_6 instead. Accordingly, the extracted $(\alpha_1, \alpha_2, G^{\varphi_1}, G^{\varphi_2}, H^t)$ fulfills Q_t with overwhelming probability assuming the hardness of the discrete-logarithm problem in $\mathbb{G}_1$.

Since we can extract $(\alpha_1, \alpha_2, G^{\varphi_1}, G^{\varphi_2}, H^t)$ for all M and (R, S, T) in θ, a question is how to find the first one with $\det(K_t) \neq 0$ if it exists. It is done as follows. Suppose that θ includes more than six valid signatures, say (R_i, S_i, T_i) for M_i for $i = 1, \ldots, q$. Given corresponding α_{ri} and α_{si} that satisfies $f_1 = 0$ and $f_2 = 0$ from (12) and (13), one can solve the equations to obtain $(u_1, u_2, u_3, v_1, v_2, v_3)$ and every t_i. Observe that, when (12) and (14) are to be

zero, we can represent α_{ri} and α_{si} by

$$\alpha_{ri} = \{(u_3 + a_3\, t_i)(v_2 + b_2\, t_i) - (v_3 + b_3\, t_i)(u_2 + a_2\, t_i)\}/\det(K_{t_i})\,, \text{ and}$$
$$\alpha_{si} = \{(v_3 + b_3\, t_i)(u_1 + a_1\, t_i) - (u_3 + a_3\, t_i)(v_1 + b_1\, t_i)\}/\det(K_{t_i}).$$

If $\det(K_{t_i}) \neq 0$, pair $(\alpha_{ri}, \alpha_{si})$ is unique to t_i. By using the extracted $(u_1, u_2, u_3, v_1, v_2, v_3)$ and t_i in each signature, we can find the smallest index $i^* \in \{1, \ldots, q\}$ at which $\det(K_{t_{i^*}}) \neq 0$ with respect to $(M_{i^*}, \Sigma_{i^*}) \in \boldsymbol{M} \times \boldsymbol{\Sigma}$, and assign ϖ accordingly. If there is no such index, we set $\varpi = \perp$. The success probability of the extraction is overwhelming since the probability of the extractor for the algebraic algorithm is overwhelming conditioned that given signatures are valid.

Case of $\Sigma \in \mathbb{G}_1 \times \mathbb{G}_2^2$.
As well as the previous case, any SPS with signature $(R, S, T) \in \mathbb{G}_1 \times \mathbb{G}_2^2$ for message $M \in \mathbb{G}_1$ verifies at least two pairing product equations that can be reduced to the following form.

$$e(R, U_1\, T^{a_1} S^{b_1})\, e(M, U_2\, T^{a_2} S^{b_2})\, e(U_3, T^{a_3})\; e(U_4, S^{b_4}) = Z_1 \tag{23}$$
$$e(R, V_1\, T^{c_1} S^{d_1})\, e(M, V_2\, T^{c_2} S^{d_2})\, e(V_3, T^{c_3})\, e(V_4, S^{d_4}) = Z_2 \tag{24}$$

Let $R = G^{\varphi_r} M^{\alpha_r}$. As before, we consider the relation in the exponent with respect to base $e(G, H)$. Then (23) and (24) are transformed as follows.

$$\{\alpha_r(u_1 + a_1 t + b_1 s) + (u_2 + a_2 t + b_2 s)\}\, m + \varphi_r(u_1 + a_1 t + b_1 s) + u_3 a_3 t + u_4 b_4 s = z_1\,, \text{ and} \tag{25}$$
$$\{\alpha_r(v_1 + c_1 t + d_1 s) + (v_2 + c_2 t + d_2 s)\}\, m + \varphi_r(v_1 + c_1 t + d_1 s) + v_3 c_3 t + v_4 d_4 s = z_2. \tag{26}$$

Consider a system of equations $Q := \{f_1 = 0, \ldots, f_5 = 0\}$ where f_i is defined as

$$f_1 = \alpha_r m + \varphi_r - r, \tag{27}$$
$$f_2 = \alpha_r(u_1 + a_1 t + b_1 s) + (u_2 + a_2 t + b_2 s), \tag{28}$$
$$f_3 = \varphi_r(u_1 + a_1 t + b_1 s) + u_3 a_3 t + u_4 b_4 s - z_1, \tag{29}$$
$$f_4 = \alpha_r(v_1 + c_1 t + d_1 s) + (v_2 + c_2 t + d_2 s)\,, \text{ and} \tag{30}$$
$$f_5 = \varphi_r(v_1 + c_1 t + d_1 s) + v_3 c_3 t + v_4 d_4 s - z_2. \tag{31}$$

Note that, with the above definition, (25) and (26) can be written as $f_2 m + f_3 = 0$ and $f_4 m + f_5 = 0$, respectively. Also note that if $u_1 + a_1 t + b_1 s \neq 0$ or $v_1 + c_1 t + d_1 s \neq 0$, then α_r is uniquely determined by Q.

Crucial Relation. For $VK = (GK, G, H, U_0, U_1, U_2, U_3, V_0, V_1, V_2, V_3)$ and $\theta = (VK, \boldsymbol{M}, \boldsymbol{\Sigma})$, let $\varpi = (\alpha_r, G^{\varphi_r}, H^s, H^t)$. Relation $\Psi(\theta, \varpi)$ returns 1 if,

- $\varpi = \perp$, and there exists (M, R, S, T) in θ for which $u_1 + a_1 t + b_1 s = 0$ and $v_1 + c_1 t + d_1 s = 0$ hold, or

for the first (M, R, S, T) in θ,

- $R = G^{\varphi_r} M^{\alpha_r}$, $S = H^s$, and $T = H^t$ hold, and
- (α_r, φ_r) determined by ϖ fulfills Q with respect to VK, M, S, and T.

In the following, we show that such ϖ is unique, useful and extractable.

Uniqueness. If θ includes a signature that causes $u_1 + a_1 t + b_1 s = 0$ and $v_1 + c_1 t + d_1 s = 0$, then ϖ must be $\perp$ to have $\Psi(\theta, \varpi) = 1$. If θ does not, then each element in ϖ is uniquely determined from the first (M, R, S, T) in θ.

Usefulness. Given $\varpi = (\alpha_r, G^{\varphi_r}, H^s, H^t)$, pick random $M^\star$ and compute $R^\star = G^{\varphi_r} M^{\star \alpha_r}$, and set $S^\star = H^s$ and $T^\star = H^t$. Then $(M^\star, R^\star, S, T)$ is a valid forgery. If $\varpi = \perp$ and $\Psi(\theta, \varpi) = 1$, we show that the scheme is insecure. Suppose that $(u_1 + a_1 t + b_1 s = 0 \wedge v_1 + c_1 t + d_1 s = 0)$ happens with respect to (M, R, S, T) in θ. From (28) and (30), we have $u_2 + a_2 t + b_2 s = 0$ and $v_2 + c_2 t + d_2 s = 0$. It results in $U_2 T^{a_2} S^{b_2} = V_2 T^{c_2} S^{d_2} = 1$ in (23) and (24). Thus, $(M^\star, R, S, T)$ is a valid forgery.

Extractability. Given (M, R, S, T), relation $(u_1 + a_1 t + b_1 s = 0 \wedge v_1 + c_1 t + d_1 s = 0)$ can be verified by testing $(U_1 T^{a_1} S^{b_1} = 1 \wedge V_1 T^{c_1} S^{d_1} = 1)$. If it happens for any signature in θ, set $\varpi = \perp$. Suppose, without loss of generality, $u_1 + a_1 t + b_1 s \neq 0$ holds. Let (M, R, S, T) be the first signature in θ. For any algebraic algorithm that outputs (R, S, T) for given M, there exists an extractor that outputs α_r such that $R = G^{\varphi_r} M^{\alpha_r}$ for some φ_r. As argued before, this α_r fulfills Q except for negligible probability if the discrete-logarithm problem in $\mathbb{G}_1$ is hard. Thus outputting $\varpi = (\alpha_r, G^{\varphi_r}, S, T)$ completes the extraction.

5 Conclusion and Open Problems

Some ideas are suggested to get around our impossibility result. The first is to resort to interactive assumptions as done for constructing 3-element scheme in [3]. The second would be to go beyond the group-preserving algebraic reduction. It however needs a number theoretic breakthrough to exploit an adversary that works for a group with different prime order. More exotic approach is to find a non-blackbox reduction that uses the adversary in non-blackbox manner. It also needs a breakthrough technique to exploit the code of the adversary to handle number-theoretic object like bilinear groups.

While this paper focused on particular type of bilinear groups due to its importance, it is of interest to see whether similar result is obtained in other settings. Since known 4-element schemes based on non-interactive assumptions only sign messages in either of the base groups but not both, it would be worth pursuing a 4-element scheme that signs group elements from both groups at the same time, or to show the impossibility.

Acknowledgements

The first author thanks Takahiro Matsuda and Yutaka Kawai for their valuable comments on an early draft of this paper. Thanks also to Fumitaka Hoshino for discussions on the generation of bilinear groups. We are grateful to reviewers in Asiacrypt'11 for their instructive comments.

References

1. M. Abe, S. S. M. Chow, K. Haralambiev, and M. Ohkubo. Double-trapdoor anonymous tags for traceable signatures. In *ACNS 2011*, LNCS 6715, pp. 183–200. Springer-Verlag, 2011.
2. M. Abe, G. Fuchsbauer, J. Groth, K. Haralambiev, and M. Ohkubo. Structure-preserving signatures and commitments to group elements. In *CRYPTO '10*, LNCS 6223, pp. 209–237. Springer, 2010.
3. M. Abe, J. Groth, K. Haralambiev, and M. Ohkubo. Optimal structure-prserving signatures in asymmetric bilinear groups. In *CRYPTO '11*, LNCS. Springer, 2011.
4. M. Abe and M. Ohkubo. A framework for universally composable non-committing blind signatures. In *ASIACRYPT 2009*, LNCS 5912, pp. 435–450. Springer, 2009.
5. B. Barak. How to go beyond the black-box simulation barrier. In *FOCS 2001*, pp. 106–115. 2001.
6. M. Bellare and A. Palacio. The knowledge-of-exponent assumptions and 3-round zero-knowledge protocols. In *CRYPTO 2004*, LNCS 3152, pp. 273–289. Springer, 2004.
7. D. Boneh and R. Venkatesan. Breaking RSA may not be equivalent to factoring. In *EUROCRYPT '98*, LNCS 1403, pp. 59–71. Springer, 1998.
8. E. Bresson, J. Monnerat, and D. Vergnaud. Separation results on the "one-more" computational problems. In *CT-RSA 2008*, LNCS 4964, pp. 71–87. Springer, 2008.
9. J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *CRYPTO '04*, LNCS 3152, pp. 56–72. Springer, 2004.
10. J. Cathalo, B. Libert, and M. Yung. Group encryption: Non-interactive realization in the standard model. In *ASIACRYPT 2009*, LNCS 5912, pp. 179–196. Springer, 2009.
11. J. Coron. Optimal security proofs for PSS and other signature schemes. In *EUROCRYPT '02*, LNCS, pp. 272–287. Springer, 2002.
12. I. Damgård. Towards practical public key systems secure against chosen ciphertext attacks. In *CRYPTO '91*, LNCS 576, pp. 445–456. Springer, 1991.
13. Y. Dodis, I. Haitner, and A. Tentes. On the (in)security of RSA signatures. ePrint 2011/087, 2011.
14. Y. Dodis, R. Oliveira, and K. Pietrzak. On the generic insecurity of the full domain hash. In *CRYPTO 2005*, LNCS 3621, pp. 449–466. Springer, 2005.
15. M. Fischlin and D. Schröder. On the impossibility of three-move blind signature schemes. In *EUROCRYPT 2010*, LNCS 6110, pp. 197–215. Springer, 2010.
16. G. Fuchsbauer. Automorphic signatures in bilinear groups. ePrint 2009/320, 2009.
17. G. Fuchsbauer. Commuting signatures and verifiable encryption. In *Eurocrypt '11*, LNCS, pp. 224–245. Springer, 2011.
18. S. Galbraith, K. Paterson, and N. Smart. Pairings for cryptographers. ePrint 2006/165, 2006.

19. S. Garg, R. Bhaskar, and S. V. Lokam. Improved bounds on security reductions for discrete log based signatures. In *CRYPTO 2008*, LNCS 5157, pp. 93–107. Springer, 2008.
20. R. Gennaro, Y. Gertner, J. Katz, and L. Trevisan. Bounds on the efficiency of generic cryptographic constructions. *SIAM J. Comput.*, 35(1):217–246, 2005.
21. S. Goldwasser, S. Micali, and R. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM J. Comp.*, 17(2):281–308, April 1988.
22. M. Green and S. Hohenberger. Universally composable adaptive oblivious transfer. In *ASIACRYPT 2008*, LNCS 5350, pp. 179–197. Springer, 2008.
23. J. Groth. Simulation-sound nizk proofs for a practical language and constant size group signatures. In *ASIACRYPT 2006*, LNCS 4284, pp. 444–459. Springer, 2006.
24. J. Groth and A. Sahai. Efficient non-interactive proof systems for bilinear groups. In *Eurocrypt '08*, LNCS 4965, pp. 415–432. Springer, 2008.
25. S. Hada and T. Tanaka. On the existence of 3-round zero-knowledge protocols. In *CRYPTO '98*, LNCS 1462, pp. 354–369. Springer, 1998. Full version available from IACR e-print archive 1999/009.
26. R. Impagliazzo and S. Rudich. Limits on the provable consequences of one-way permutations. In *STOC'89*, pp. 44–61. ACM, 1989.
27. J. Katz, D. Schröder, and A. Yerukhimovich. Impossibility of blind signatures from one-way permutations. In *TCC 2011*, LNCS 6597, pp. 615–629. Springer, 2011.
28. T. Malkin, R. Moriarty, and N. Yakovenko. Generalized environmental security from number theoretic assumptions. In *TCC 2006*, LNCS 3876, pp. 343–359. Springer, 2006.
29. P. Paillier. Impossibility proofs for RSA signatures in the standard model. In *CT-RSA 2007*, LNCS 4377, pp. 31–48. Springer, 2007.
30. P. Paillier and D. Vergnaud. Discrete-log-based signatures may not be equivalent to discrete log. In *ASIACRYPT 2005*, LNCS 3788, pp. 1–20. Springer, 2005.
31. P. Paillier and J. L. Villar. Trading one-wayness against chosen-ciphertext security in factoring-based encryption. In *ASIACRYPT 2006*, LNCS 4284, pp. 252–266. Springer, 2006.
32. R. Pass. Limits of provable security from standard assumptions. In *STOC 2011*, pp. 109–118. ACM, 2011.
33. O. Reingold, L. Trevisan, and S. P. Vadhan. Notions of reducibility between cryptographic primitives. In *TCC 2004*, LNCS 2951, pp. 1–20. Springer, 2004.
34. D. R. Simon. Finding collisions on a one-way street: Can secure hash functions be based on general assumptions? In *EUROCRYPT '98*, LNCS 1403, pp. 334–345. Springer, 1998.