# Towards a Tightly Secure Signature in Multi-User Setting with Corruptions Based on Search Assumptions

Hirofumi Yoshioka[1], Wakaha Ogata[1], and Keitaro Hashimoto[2]

[1]Tokyo Institute of Technology
fumisket@gmail.com, ogata.w.aa@m.titech.ac.jp
[2]National Institute of Advanced Industrial Science and Technology (AIST)
keitaro.hashimoto@aist.go.jp

**Abstract**

This paper is a report on how we tackled constructing a digital signature scheme whose *multi-user security with corruption* can be *tightly* reduced to *search assumptions*. We fail to (dis)prove the statement but obtain the following new results:

- We reveal two new properties of signature schemes whose security cannot be tightly reduced to standard assumptions.
- We construct a new signature scheme. Its multi-user security with corruption is reduced to the CDH assumption (in the ROM), and its reduction loss is independent of the number of users but depends on the number of RO queries.

## 1 Introduction

This paper is a report on how we tackled constructing a digital signature scheme whose *multi-user security with corruption* can be *tightly* reduced to *search assumptions*. As summarized in Table 1, all known tightly-secure signatures in multi-user security with corruption are based on interactive search assumptions [WLG+19] or decision assumptions [GJ18, Bad14, BHJ+15, ABP19, DGJL21]. Thus, such a signature based on (non-interactive) search problems remains open. We tackle this problem and obtain the following results. First, we reveal two new properties of signature schemes whose security cannot be tightly reduced to standard assumptions. More precisely, we generalize the negative result of Pan and Wagner [PW22], which shows that the reduction loss of the concrete signature scheme, called the Parallel-OR signature scheme, is lower bounded by the number of users. From this negative result, we have precious knowledge about designing a signature scheme that is tightly secure in multi-user settings with corruption. Next, we show a concrete construction of signature schemes based on the first result. Our scheme's multi-user security can be reduced to the CDH assumption, and the reduction loss does not depend on the number of users, but, unfortunately, the loss linearly depends on the number of random oracle queries issued by the adversary. So, it remains open whether we can construct a signature scheme whose multi-user security with corruption can be tightly reduced to search assumptions.

## 2 Preliminaries

**Notations.** Let $\lambda \in \mathbb{N}$ be a security parameter. For natural number $N$, let $[N] := \{1, 2, \ldots, N\}$. For an algorithm $\mathsf{X}$ and its input $x$, let $\mathsf{X}(x)$ be the set of all output. For random variables $X$ and $Y$, $\mathsf{SD}(X; Y)$

Table 1: Existing tightly secure signatures.

| Scheme | Security model | Computational assumption |
|---|---|---|
| [GJ03, Che05, KLP17] | Single-user | CDH |
| [PR20] | Multi-user w/o Corruption | CDH |
| [WLG+19] | Multi-user w/ Corruption | One-more CDH |
| [Bad14] | Multi-user w/ Corruption | SXDH |
| [BHJ+15, ABP19] | Multi-user w/ Corruption | DLIN |
| [GJ18] | Multi-user w/ Corruption | CDH+DDH |
| [DGJL21, PW22] | Multi-user w/ Corruption | DDH |

denotes the statistical distance between them.

**Computational assumption.** Let $\mathbb{G}$ be a multiplicative group with prime order $p$ and $g \in \mathbb{G}$ be its generator. We say that the computational Diffie-Hellman (CDH) assumption holds in $\mathbb{G}$ if for any ppt adversary, the advantage defined by the following is negligibly small.

$$\mathsf{Adv}^{\mathsf{CDH}}_{\mathcal{A}}(\lambda) := \Pr[Z = g^{xy} : x, y \leftarrow\!\!\$\, \mathbb{Z}_p; Z \leftarrow \mathcal{A}(g, g^x, g^y)].$$

**Digital signature.** A digital signature scheme $\mathsf{SIG} = (\mathsf{Setup}, \mathsf{Gen}, \mathsf{Sig}, \mathsf{Ver})$ is defined as follows.

- $\mathsf{Setup}(1^\lambda)$, taking the security parameter $\lambda$ as an input, generates a system parameter $\mathsf{par}$, which describes spaces of public keys $K_p$, secret keys $K_s$, messages $M$ and signatures $S$. We may omit $\mathsf{par}$ as input in the following algorithms.
- $\mathsf{Gen}(\mathsf{par})$ generates a pair of a public key and a secret key $(\mathsf{pk}, \mathsf{sk}) \in K_p \times K_s$.
- $\mathsf{Sig}(\mathsf{sk}, \mathsf{m})$, taking a secret key $\mathsf{sk}$ and message $\mathsf{m} \in M$, computes a signature $\sigma \in S$.
- $\mathsf{Ver}(\mathsf{pk}, \mathsf{m}, \sigma)$, taking a public key $\mathsf{pk}$, message $\mathsf{m}$, and a signature $\sigma$, outputs a bit $b \in \{0, 1\}$.

A signature scheme is said to be correct[1] if for any $\lambda \in \mathbb{N}$, $\mathsf{par} \in \mathsf{Setup}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \in \mathsf{Gen}(\mathsf{par})$, $\mathsf{m} \in M$, and $\sigma \in \mathsf{Sig}(\mathsf{sk}, \mathsf{m})$, $\mathsf{Ver}(\mathsf{pk}, \mathsf{m}, \sigma)$ always outputs 1.

For a public key $\mathsf{pk}$, we define the following set:

$$SK(\mathsf{pk}) := \{\mathsf{sk} \mid (\mathsf{pk}, \mathsf{sk}) \in \mathsf{Gen}(\mathsf{par})\}.$$

Multi-user security with adaptive corruption of signature schemes is defined as follows.

**Definition 1** (Multi-user security [PW22])**.** *For a signature scheme* $\mathsf{SIG}$*, consider a game* $N$-**MU-UF-CMA-C** *shown in Algorithm 1. We say* $\mathsf{SIG}$ *has* $N$-MU-UF-CMA-C *security if for any ppt adversary* $\mathcal{A}$*, the advantage*

$$\mathsf{Adv}^{N\text{-}\mathsf{MU\text{-}UF\text{-}CMA\text{-}C}}_{\mathcal{A},\mathsf{SIG}}(\lambda) := \Pr[N\text{-}\mathbf{MU\text{-}UF\text{-}CMA\text{-}C}^{\mathcal{A}}_{\mathsf{SIG}}(\lambda) \Rightarrow 1]$$

*in negligibly small.*

As in [PW22], we introduce the following weaker security notion, multi-user security with *static* corruption *without signing oracle*. We note that impossibility results in the weaker security notion imply that in the stronger notion.

**Definition 2** (Static security [PW22])**.** *For signature scheme* $\mathsf{SIG}$*, consider a game* $N$-**MU-UF-S** *shown in Algorithm 2. If for any ppt adversary* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ *the advantage*

$$\mathsf{Adv}^{N\text{-}\mathsf{MU\text{-}UF\text{-}S}}_{\mathcal{A},\mathsf{SIG}}(\lambda) := \Pr[N\text{-}\mathbf{MU\text{-}UF\text{-}S}^{\mathcal{A}}_{\mathsf{SIG}}(\lambda) \Rightarrow 1]$$

*in negligibly small, we say* $\mathsf{SIG}$ *has* $N$-MU-UF-S *security.*

**Algorithm 1** $N$-**MU-UF-CMA-C**$^{\mathcal{A}}_{\mathsf{SIG}}(\lambda)$

$\mathsf{par} \leftarrow \mathsf{Setup}(1^\lambda)$
**for** $i \in [N]$ **do** $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Gen}(\mathsf{par})$
$(i^*, \mathsf{m}^*, \sigma^*) \leftarrow \mathcal{A}^{\mathtt{Corr},\mathtt{Sig}}(\mathsf{par}, (\mathsf{pk}_i)_{i\in[N]})$
**if** $i^* \in \mathcal{L}_{id}$ **then return** $0$
**if** $\exists \sigma : (i^*, \mathsf{m}^*, \sigma) \in \mathcal{L}_m$ **then return** $0$
**return** $\mathsf{Ver}(\mathsf{pk}_{i^*}, \mathsf{m}^*, \sigma^*)$
**Oracle** $\mathtt{Corr}(i)$
$\mathcal{L}_{id} := \mathcal{L}_{id} \cup \{i\}$
**return** $\mathsf{sk}_i$
**Oracle** $\mathtt{Sig}(i, \mathsf{m})$
$\sigma \leftarrow \mathsf{Sig}(\mathsf{sk}_i, \mathsf{m})$
$\mathcal{L}_m := \mathcal{L}_m \cup \{(i, \mathsf{m}, \sigma)\}$
**return** $\sigma$

**Algorithm 2** $N$-**MU-UF-S**$^{\mathcal{A}}_{\mathsf{SIG}}(\lambda)$

$\mathsf{par} \leftarrow \mathsf{Setup}(1^\lambda)$
**for** $i \in [N]$ **do** $(\mathsf{pk}_i, \mathsf{sk}_i) \leftarrow \mathsf{Gen}(\mathsf{par})$
$(j, St_{\mathcal{A}}) \leftarrow \mathcal{A}_1(\mathsf{par}, (\mathsf{pk}_i)_{i\in[N]})$
**if** $j \notin [N]$ **then return** $0$
$(\mathsf{m}^*, \sigma^*) \leftarrow \mathcal{A}_2(St_{\mathcal{A}}, (\mathsf{sk}_i)_{i\in[N]\setminus\{j\}})$
**return** $\mathsf{Ver}(\mathsf{pk}_j, \mathsf{m}^*, \sigma^*)$

**Definition 3** (Key-pair Verifiability)**.** *If there exists a ppt algorithm* $\mathsf{VerK}$ *such that the next equation holds for any* $\lambda \in \mathbb{N}$, $\mathsf{par} \in \mathsf{Setup}(1^\lambda)$, $\mathsf{pk} \in K_p$, *and* $\mathsf{sk} \in K_s$, *then* $\mathsf{SIG}$ *is said to be key-pair verifiable.*

$$\mathsf{VerK}(\mathsf{par}, \mathsf{pk}, \mathsf{sk}) = 1 \iff (\mathsf{pk}, \mathsf{sk}) \in \mathsf{Gen}(\mathsf{par})$$

*If* $\mathsf{VerK}(\mathsf{par}, \mathsf{pk}, \mathsf{sk}) = 1$, $\mathsf{sk}$ *is a valid secret key of* $\mathsf{pk}$.

Hereafter, we assume any signature scheme has key-pair verifiability, since we can verify the validity of a given pair $(\mathsf{pk}, \mathsf{sk})$ by repeating the procedure of computing a signature of a random message using $\mathsf{sk}$ and verifying it using $\mathsf{pk}$ enough number of times.

**Non-interactive problems (NIP) and simple reductions.** Existing impossibility results [AGO11, PW22] are for *simple* reductions that reduce the security of signature schemes to *non-interactive problems*. *Non-interactive problems* (NIP) is a wide class of mathematical problems such that, given an instance of the problem, the solver needs to output an answer without accessing any oracles. This class includes both decision problems such as DDH and search problems such as DLP and CDH.[2]

**Definition 4** (Non-interactive problem [AGO11, PW22])**.** *Non-interactive problem is formalized as a tuple of algorithms* $\mathsf{NIP} = (\mathsf{T}, \mathsf{U}, \mathsf{V})$.

- $\mathsf{T}(1^\lambda)$, *taking the security parameter* $\lambda$ *as an input, outputs an instance* $c$ *and its witness* $w$.
- $\mathsf{U}(c)$ *takes an instance* $c$ *as input, and outputs a candidate of solutions* $s$.
- $\mathsf{V}(c, w, s)$ *takes* $c, w, s$ *as input, and outputs a bit.*

[1] In this paper, we only consider the perfect correctness.
[2] NIP includes both decision problems and search problems, but not one-more type problems, since the solver is given an oracle.

**Algorithm 3** $\mathbf{NIP}^{\mathsf{X}}_{\mathsf{NIP}}(\lambda)$ $\quad (\mathsf{X} \in \{\mathcal{A}, \mathsf{U}\})$

$(c, w) \leftarrow \mathsf{T}(1^\lambda)$
$s \leftarrow \mathsf{X}(c)$
**return** $\mathsf{V}(c, w, s)$

**Algorithm 4** $\mathcal{R}^{\mathcal{A}}(c)$

$(St_{\mathcal{R}}, \mathsf{par}, (\mathsf{pk}_i)_{i\in[N]}) \leftarrow \mathcal{R}_1(c)$
$(j, St_{\mathcal{A}}) \leftarrow \mathcal{A}_1^{\mathsf{H}}(\mathsf{par}, (\mathsf{pk}_i)_{i\in[N]})$
$(St_{\mathcal{R}}, (\mathsf{sk}_i)_{i\in[N]\setminus\{j\}}) \leftarrow \mathcal{R}_2(St_{\mathcal{R}}, j)$
$(\mathsf{m}^*, \sigma^*) \leftarrow \mathcal{A}_2^{\mathsf{H}}(St_{\mathcal{A}}, (\mathsf{sk}_i)_{i\in[N]\setminus\{j\}})$
**return** $\mathcal{R}_3(St_{\mathcal{R}}, j, \mathsf{m}^*, \sigma^*)$
**Oracle** $\mathsf{H}(\mathsf{query})$
$(St_{\mathcal{R}}, h) \leftarrow \mathcal{R}_{\mathrm{RO}}(St_{\mathcal{R}}, \mathsf{query})$
**return** $h$

*Consider the game* **NIP** *depicted in Algorithm 3. For an algorithm $\mathcal{A}$, its advantage is defined as*

$$\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{A}}(\lambda) := \left| \Pr[\mathbf{NIP}^{\mathcal{A}}_{\mathsf{NIP}}(\lambda) \Rightarrow 1] - \Pr[\mathbf{NIP}^{\mathsf{U}}_{\mathsf{NIP}}(\lambda) \Rightarrow 1] \right|.$$

*If the advantage is negligibly small for any ppt algorithms $\mathcal{A}$, we say* NIP *is hard.*

Roughly speaking, a *simple* reduction is a reduction that has black-box access to the adversary algorithm $\mathcal{A}$ only once and without rewinding. In this paper, we only deal with simple reductions that reduce the $N$-MU-UF-S security of signature schemes to an NIP.

**Definition 5** (Simple reduction [PW22])**.** *A simple* $(\mathsf{NIP}, \mathsf{SIG})$*-reduction $\mathcal{R} = (\mathcal{R}_1, \mathcal{R}_2, \mathcal{R}_3, \mathcal{R}_{\mathrm{RO}})$ is a tuple of algorithms to solve* NIP*, having a black-box access to $\mathcal{A}$ only once, where $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ is an adversary against* SIG*'s $N$-MU-UF-S security. Without loss of generality, we assume that only $\mathcal{R}_1$ is a probabilistic algorithm, and $\mathcal{R}_2$, $\mathcal{R}_3$, $\mathcal{R}_{\mathrm{RO}}$ are deterministic.*

- *$\mathcal{R}_1(c)$ receives an instance $c$ of* NIP*, and outputs own state information $St_{\mathcal{R}}$, parameters* par *of the signature scheme, and a list of public keys $(\mathsf{pk}_i)_{i\in[N]}$.*
- *$\mathcal{R}_2(St_{\mathcal{R}}, j)$ receives an index $j \in [N]$ from $\mathcal{A}$ addition to the current state $St_{\mathcal{R}}$, and outputs a new state $St_{\mathcal{R}}$ and a list of secret keys $(\mathsf{sk}_i)_{i\in[N]\setminus\{j\}}$.*
- *$\mathcal{R}_3(St_{\mathcal{R}}, j, \mathsf{m}^*, \sigma^*)$ receives $j \in [N]$, $\mathsf{m}^*$, $\sigma^*$ from $\mathcal{A}$ as well as the current state. It outputs a solution $s$ of the instance $c$ of* NIP*.*
- *$\mathcal{R}_{\mathrm{RO}}(St_{\mathcal{R}}, \mathsf{query})$ receives* query *and the current state. It outputs a new state $St_{\mathcal{R}}$ and a hash value $h$.*

*Algorithm 4 shows the interaction between $\mathcal{R}$ and $\mathcal{A}$. Let $V_{\mathcal{R}}$ and $V_{real}$ be random variables representing $\mathcal{A}$'s view interacting with $\mathcal{R}$ and that interacting with the challenger in $N$-***MU-UF-S** *game, respectively. For a function $L$, we say that $\mathcal{R}$ is $(N, \delta_{\mathcal{R}}, L)$-simple if*

$$\mathsf{SD}(V_{\mathcal{R}}; V_{real}) \leq \delta_{\mathcal{R}},$$
$$\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{R}^{\mathcal{A}}}(\lambda) \geq L(\lambda, N, \mathsf{Adv}^{N\text{-MU-UF-S}}_{\mathcal{A},\mathsf{SIG}}(\lambda))$$

*holds for any ppt adversary $\mathcal{A}$.*

# 3 New Impossibility Results

## 3.1 Preparation

First, we introduce two new properties of digital signature schemes $\mathsf{SIG}$.

**Definition 6** (Signature statistically close)**.** *Let* $SIG(\mathsf{sk}, \mathsf{m})$ *be a random variable representing the output of* $\mathsf{Sig}(\mathsf{sk}, \mathsf{m})$*.* $\mathsf{SIG}$ *is said to be* $\varepsilon_{\mathsf{Sig}}$*-signature statistically close if for any* $\mathsf{m} \in M$*,* $\mathsf{pk} \in K_p$ *and two valid secret keys* $\mathsf{sk}, \mathsf{sk}' \in SK(\mathsf{pk})$*, it holds that*

$$\mathsf{SD}(SIG(\mathsf{sk}, \mathsf{m}); SIG(\mathsf{sk}', \mathsf{m})) \leq \varepsilon_{\mathsf{Sig}}.$$

**Definition 7** (RO statistically close)**.** *Let* $Q(\mathsf{sk}, \mathsf{m})$ *be a random variable representing the random oracle queries issued in the run of* $\mathsf{Sig}^{\mathsf{H}}(\mathsf{sk}, \mathsf{m})$*.* $\mathsf{SIG}$ *is said to be* $\varepsilon_{\mathrm{RO}}$-RO *statistically close if for any* $\mathsf{m} \in M$*,* $\mathsf{pk} \in K_p$ *and two valid secret keys* $\mathsf{sk}, \mathsf{sk}' \in SK(\mathsf{pk})$*, it holds that*

$$\mathsf{SD}(Q(\mathsf{sk}, \mathsf{m}); Q(\mathsf{sk}', \mathsf{m})) \leq \varepsilon_{\mathrm{RO}}.$$

## 3.2 Our Impossibility Results

By using the above properties, we obtain the following impossibility result.

**Theorem 1.** *Let* $\mathsf{SIG}$ *be a* $\varepsilon_{\mathsf{Sig}}$*-signature statistically close and* $\varepsilon_{\mathrm{RO}}$-RO *statistically close signature scheme. For any* $(N, \delta_{\mathcal{R}}, L)$*-simple* $(\mathsf{NIP}, \mathsf{SIG})$*-reduction* $\mathcal{R}$*, there exists an algorithm* $\mathcal{M}$ *that solves* $\mathsf{NIP}$ *such that*

$$\begin{aligned} \mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}}(\lambda) &\geq L(\lambda, N, 1) - (4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}) - 1/N, \\ \mathbf{T}(\mathcal{M}) &\leq N \cdot \mathbf{T}(\mathcal{R}) + N(N-1)\mathbf{T}(\mathsf{VerK}) + \mathbf{T}(\mathsf{Sig}), \end{aligned}$$

*where* $\mathbf{T}(X)$ *denotes the running time of* $X$*.*

*Proof.* The proof proceeds in almost the same way as the impossibility proof of [BJLS16, PW22], that is, we first construct a computationally-unbounded adversary $\mathcal{A}_\infty$ who wins the $N$-**MU-UF-S** game with probability 1 and then construct a meta-reduction $\mathcal{M}$ that solves $\mathsf{NIP}$ by efficiently simulating the adversary $\mathcal{A}_\infty$ against the reduction $\mathcal{R}$. The difference from the existing proofs is how to efficiently simulate $\mathcal{A}_\infty$: [BJLS16] uses key-rerandomizability and [PW22] uses properties of Parallel-OR signatures to show that $\mathcal{M}$'s simulation of $\mathcal{A}_\infty$ does not significantly change the output of $\mathcal{R}$. In this work, we prove the same statement using RO statistically close and signature statistically close of $\mathsf{SIG}$.

$\mathcal{A}_\infty = (\mathcal{A}_{\infty,1}, \mathcal{A}_{\infty,2})$ is described in Algorithm 5. $\mathcal{A}_\infty$ finds a secret key for $\mathsf{pk}_{j^*}$ with brute-forces search. Note that this process makes $\mathcal{A}_\infty$ inefficient. Then, it generates a forged signature by signing a randomly chosen message with the founded secret key. By the definition of $\mathcal{A}_\infty$, the following holds.

**Lemma 1.** $\mathsf{Adv}^{N\text{-}\mathsf{MU\text{-}UF\text{-}S}}_{\mathcal{A}_\infty, \mathsf{SIG}}(\lambda) = 1$.

Next, we consider a series of meta-reductions $\mathcal{M}_1, \mathcal{M}_2, \mathcal{M}_3 (= \mathcal{M})$ that try to solve $\mathsf{NIP}$ using $\mathcal{R}$ shown in Algorithms 7 to 9. As shown below, the behavior of the meta-reduction is modified in the order to efficiently simulate the behavior of the adversary $\mathcal{A}_\infty$ in the end.

- $\mathcal{M}_1$ interacts with $\mathcal{R}$ in the same way as $\mathcal{A}_\infty$, except for $\mathsf{Rewind}$ step. The subroutine $\mathsf{Rewind}$ shown in Algorithm 6 executes $\mathcal{R}_1$ and receives its output $(St_{\mathcal{R},1}, \mathsf{par}, (\mathsf{pk}_i)_{i\in[N]})$. Then it executes $\mathcal{R}_2$ independently for all $j \in [N]$ and sets the $\mathsf{succ}[j]$ flag if all secret keys returned by $\mathcal{R}_2$ are valid. If $\mathsf{succ}[j]$ is set, the obtained secret keys are stored (but they are not used by $\mathcal{M}_1$). After $\mathsf{Rewind}$ step, $\mathcal{M}_1$ randomly selects $j^*$ as in $\mathcal{A}_\infty$, executes $\mathcal{R}_3$ using state $St_{\mathcal{R},2,j^*}$ (i.e., as a continuation of the $j^*$-th $\mathcal{R}_2$ execution), and returns the output of $\mathcal{R}_3$ as its output.

**Algorithm 5** $\mathcal{A}^{\mathsf{H}}_{\infty} = (\mathcal{A}^{\mathsf{H}}_{\infty,1}, \mathcal{A}^{\mathsf{H}}_{\infty,2})$

$\mathcal{A}^{\mathsf{H}}_{\infty,1}(\mathsf{par}, (\mathsf{pk}_i)_{i\in[N]})$

$j^* \xleftarrow{\$} [N]$
$St := (\mathsf{par}, (\mathsf{pk}_i)_{i\in[N]}, j^*)$
**return** $(j^*, St)$

$\mathcal{A}^{\mathsf{H}}_{\infty,2}(St, (\mathsf{sk}_i)_{i\in[N]\setminus\{j^*\}})$

**if** $\exists i \in [N] \setminus \{j^*\} : \mathsf{VerK}(\mathsf{par}, \mathsf{pk}_i, \mathsf{sk}_i) = 0$ **then**
    **return** $\bot$
$\mathsf{sk} \xleftarrow{\$} SK(\mathsf{pk}_{j^*})$ ▷ Brute-force search
$\mathsf{m}^* \xleftarrow{\$} M$
$\sigma^* \leftarrow \mathsf{Sig}(\mathsf{sk}, \mathsf{m}^*)$
**return** $(\mathsf{m}^*, \sigma^*)$

**Algorithm 6** $\mathsf{Rewind}^{\mathcal{R}}(c)$

$\rho_{\mathcal{R}} \xleftarrow{\$} \{0,1\}^z$
$(St_{\mathcal{R},1}, \mathsf{par}, (\mathsf{pk}_i)_{i\in[N]}) \leftarrow \mathcal{R}_1(c; \rho_{\mathcal{R}})$
**for** $j \in [N]$ **do**
    $(St_{\mathcal{R},2,j}, (\mathsf{sk}_i)_{i\in[N]\setminus\{j\}}) \leftarrow \mathcal{R}_2(St_{\mathcal{R},1}, j)$
    $\mathsf{succ}[j] := 1$
    **for** $i \in [N] \setminus \{j\}$ **do**
        **if** $\mathsf{VerK}(\mathsf{par}, \mathsf{pk}_i, \mathsf{sk}_i) = 0$ **then** $\mathsf{succ}[j] := 0$
    **if** $\mathsf{succ}[j] = 1$ **then**
        **for** $i \in [N] \setminus \{j\}$ **do** $\mathsf{sk}[i] := \mathsf{sk}_i$
**return** $(\mathsf{par}, \mathsf{succ}[\cdot], \mathsf{sk}[\cdot], (St_{\mathcal{R},2,j})_{j\in[N]})$

- $\mathcal{M}_2$ is the same as $\mathcal{M}_1$ except it additionally checks whether **bad** event occurs, and halts if the event occurs. **bad** event occurs if $\mathsf{succ}[j^*] = 1$ and for all $j \neq j^*$, $\mathsf{succ}[j] = 0$. That is, this event means $\mathcal{R}$ succeeds in simulating the corruption oracle only for the $j^*$-th run of $\mathcal{R}_2$ but fails for all of the other runs. Therefore, if the **bad** event does not occur, a valid secret key of $j^*$-th user is obtained in the $\mathsf{Rewind}$ step for $j(\neq j^*)$.
- $\mathcal{M}_2$ is the same as $\mathcal{M}_2$ except that it uses the secret key obtained in the $\mathsf{Rewind}$ step to forge a signature instead of searching it brute-force. Here, $\mathcal{M}_3$ is efficient because it no longer finds a secret key with an inefficient brute-force search.

We can show that the above modifications of meta-reductions do not significantly change $\mathcal{R}$'s advantage.

**Algorithm 7** $\mathcal{M}_1(c)$

$(\mathsf{par}, \mathsf{succ}[\cdot], \mathsf{sk}[\cdot], (St_{\mathcal{R},2,j})_{j\in[N]}) \leftarrow \mathsf{Rewind}^{\mathcal{R}}(c)$
$j^* \xleftarrow{\$} [N]$
**if** $\mathsf{succ}[j^*] \neq 1$ **then return** 0
$\mathsf{sk} \xleftarrow{\$} SK(\mathsf{pk}_{j^*})$ ▷ Brute-force search
$\mathsf{m}^* \xleftarrow{\$} M$
$\sigma^* \leftarrow \mathsf{Sig}(\mathsf{sk}, \mathsf{m}^*)$
**return** $\mathcal{R}_3(St_{\mathcal{R},2,j^*}, j^*, \mathsf{m}^*, \sigma^*)$

**Algorithm 8** $\mathcal{M}_2(c)$

$(\mathsf{par}, \mathsf{succ}[\cdot], \mathsf{sk}[\cdot], (St_{\mathcal{R},2,j})_{j\in[N]}) \leftarrow \mathsf{Rewind}^{\mathcal{R}}(c)$
$j^* \xleftarrow{\$} [N]$
**if** $\mathsf{succ}[j^*] \neq 1$ **then return** $0$
**if** $\forall j \in [N]\backslash\{j^*\} : \mathsf{succ}[j] = 0$ **then**
    $\mathbf{bad} := 1$
    **return** $\perp$
$\mathsf{sk} \xleftarrow{\$} SK(\mathsf{pk}_{j^*})$ ▷ Brute-force search
$\mathsf{m}^* \xleftarrow{\$} M$
$\sigma^* \leftarrow \mathsf{Sig}(\mathsf{sk}, \mathsf{m}^*)$
**return** $\mathcal{R}_3(St_{\mathcal{R},2,j^*}, j^*, \mathsf{m}^*, \sigma^*)$

**Algorithm 9** $\mathcal{M}_3(c)$

$(\mathsf{par}, \mathsf{succ}[\cdot], \mathsf{sk}[\cdot], (St_{\mathcal{R},2,j})_{j\in[N]}) \leftarrow \mathsf{Rewind}^{\mathcal{R}}(c)$
$j^* \xleftarrow{\$} [N]$
**if** $\mathsf{succ}[j^*] \neq 1$ **then return** $0$
**if** $\forall j \in [N]\backslash\{j^*\} : \mathsf{succ}[j] = 0$ **then**
    $\mathbf{bad} := 1$
    **return** $\perp$
$\mathsf{sk}_{\mathcal{R}} := \mathsf{sk}[j^*]$
$\mathsf{m}^* \xleftarrow{\$} M$
$\sigma^* \leftarrow \mathsf{Sig}(\mathsf{sk}_{\mathcal{R}}, \mathsf{m}^*)$
**return** $\mathcal{R}_3(St_{\mathcal{R},2,j^*}, j^*, \mathsf{m}^*, \sigma^*)$

**Lemma 2.** $\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{R}^{\mathcal{A}_\infty}}(\lambda) = \mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}_1}(\lambda)$.

*Proof.* As observed, the output of $\mathcal{M}_1$ is independent of the execution of $\mathcal{R}_2(St_{\mathcal{R},1}, j)$ against $j \neq j^*$ in the $\mathsf{Rewind}$ step. Therefore, the execution of $\mathcal{R}_2(St_{\mathcal{R},1}, j)$ against $j \neq j^*$ in the $\mathsf{Rewind}$ step never affects the output of $\mathcal{R}_3$ and thus the output of $\mathcal{M}_1$. With this in mind, and given that the conditions for the line 4 of the algorithm 5 and the line 3 of the algorithm 7 are identical, the output of $\mathcal{R}^{\mathcal{A}_\infty}$ and the outputs of $\mathcal{M}_1$ are equivalent. □

**Lemma 3.** $\left|\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}_1}(\lambda) - \mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}_2}(\lambda)\right| \leq 1/N$.

*Proof.* The difference between $\mathcal{M}_1$ and $\mathcal{M}_2$ is the behavior when the **bad** event occurs. This event occurs when $\mathcal{R}_2$ succeeds in simulating the corruption oracle for exactly one index, and the index is $j^*$. Since $j^*$ is chosen uniformly at random from $[N]$, the lemma holds. □

**Lemma 4.** $\left|\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}_2}(\lambda) - \mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}_3}(\lambda)\right| \leq 4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}$.

*Proof.* The difference between $\mathcal{M}_2$ and $\mathcal{M}_3$ is the secret key used to generate the forged signature; $\mathcal{M}_2$ uses the secret key $\mathsf{sk}$ sampled uniformly at random from $SK(\mathsf{pk}_{j^*})$, but $\mathcal{M}_3$ uses the secret key $\mathsf{sk}_{\mathcal{R}}$ received from $\mathcal{R}_2$ during the $\mathsf{Rewind}$ step. Recall that if **bad** does not occur, $\mathsf{sk}[j^*] = \mathsf{sk}_{\mathcal{R}}$ is a valid secret key corresponding to $\mathsf{pk}_{j^*}$.

The information $\mathcal{R}$ can see in the interaction with $\mathcal{M}_2$ is $\mathsf{m}^* \leftarrow M$, $\sigma^* \leftarrow \mathsf{Sig}(\mathsf{sk}, \mathsf{m}^*)$ and the RO query sequence $Q(\mathsf{sk}, \mathsf{m}^*)$, observed by $\mathcal{R}_{\mathrm{RO}}$, which is generated in the consequence of $\mathsf{Sig}(\mathsf{sk}, \mathsf{m}^*)$. On the other hand, the information $\mathcal{R}$ can see in the interaction with $\mathcal{M}_3$ is $\mathsf{m}^* \leftarrow M$, $\sigma^* \leftarrow \mathsf{Sig}(\mathsf{sk}_{\mathcal{R}}, \mathsf{m}^*)$, and $Q(\mathsf{sk}_{\mathcal{R}}, \mathsf{m}^*)$.

First, let us consider the case $\mathcal{R}$ perfectly simulates the $N$-**MU-UF-S** game (i.e., the case $\delta_{\mathcal{R}} = 0$). Observe that the distribution of the forged signature created by $\mathcal{M}_2$ is identical to the distribution of signatures generated by $\mathsf{Sig}(\mathsf{sk}, \cdot)$, and the distribution of the forged signature created by $\mathcal{M}_3$ is identical to the distribution of signatures generated by $\mathsf{Sig}(\mathsf{sk}_{\mathcal{R}}, \cdot)$. Since $\mathsf{SIG}$ is $\varepsilon_{\mathsf{Sig}}$-signature statistically close, the statistical distance between the forged signature created by $\mathcal{M}_2$ and the one created by $\mathcal{M}_3$ is less than $\varepsilon_{\mathsf{Sig}}$. Moreover, the RO query sequences $Q(\mathsf{sk}, \mathsf{m}^*)$ and $Q(\mathsf{sk}_{\mathcal{R}}, \mathsf{m}^*)$ have a statistical distance of at most $\varepsilon_{\mathrm{RO}}$ due to $\varepsilon_{\mathrm{RO}}$-RO statistically close of $\mathsf{SIG}$. Therefore, the statistical distance between the final output of $\mathcal{R}_3$ interacting with $\mathcal{M}_2$ and interacting with $\mathcal{M}_3$ is at most $\varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}$.

Now, let us consider the general case where $\delta_{\mathcal{R}} > 0$. In this case, the distribution of the forged signature created by $\mathcal{M}_2$ may differ from the distribution of signatures generated by $\mathsf{Sig}(\mathsf{sk}, \cdot)$ (due to e.g., "biased" RO simulation). However, the statistical distance between $\mathcal{R}$'s simulation and the real game is at most $\delta_{\mathcal{R}}$ due to Definition 5. Thus, the statistical distance between the distribution of the forged signature created by $\mathcal{M}_2$ and the distribution of signatures generated by $\mathsf{Sig}(\mathsf{sk}, \cdot)$ is at most $\delta_{\mathcal{R}}$. Similarly, the statistical distance between the distribution of the forged signature created by $\mathcal{M}_3$ and the distribution of signatures generated by $\mathsf{Sig}(\mathsf{sk}_{\mathcal{R}}, \cdot)$ is at most $\delta_{\mathcal{R}}$. Therefore, the statistical distance between the forged signatures created by $\mathcal{M}_2$ and the one created by $\mathcal{M}_3$ is at most $2\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}}$. Similarly, the statistical distance of the RO query sequences is at most $2\delta_{\mathcal{R}} + \varepsilon_{\mathrm{RO}}$. Therefore, the total statistical distance between $\mathcal{M}_2$'s and $\mathcal{M}_3$'s simulation is at most $4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}$. □

From Definition 5 and Lemma 1–4, we obtain

$$\begin{aligned}
\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}_3}(\lambda) &\geq \mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{R}^{\mathcal{A}_\infty}}(\lambda) - (4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}) - \frac{1}{N} \\
&\geq L(\lambda, N, \mathsf{Adv}^{N\text{-}\mathsf{MU\text{-}UF\text{-}S}}_{\mathcal{A}_\infty, \mathsf{SIG}}(\lambda)) - (4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}) - \frac{1}{N} \\
&= L(\lambda, N, 1) - (4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}) - \frac{1}{N}.
\end{aligned}$$

The running time of $\mathcal{M}_3$ can be evaluated as

$$\mathbf{T}(\mathcal{M}_3) \leq N \cdot \mathbf{T}(\mathcal{R}) + N(N-1)\mathbf{T}(\mathsf{VerK}) + \mathbf{T}(\mathsf{Sig}).$$

This concludes the proof. □

From Theorem 1, the reduction loss from the multi-user security to an NIP is lower bounded by the number of users $N$, if $\varepsilon_{\mathsf{Sig}}, \varepsilon_{\mathrm{RO}}, \delta_{\mathcal{R}}$ are negligibly small.

### 3.3 Generalization

Similarly to [BJLS16, Theorem 4], Theorem 1 can be generalized for $r$-simple reduction that is allowed to rewind $\mathcal{A}$ $r$ times sequentially. The lower bound is preserved for generalized reductions.

**Theorem 2.** *Let* SIG *be a* $\varepsilon_{\mathsf{Sig}}$*-signature statistically close and* $\varepsilon_{\mathrm{RO}}$-RO *statistically close signature scheme. For any* $(N, \delta_{\mathcal{R}}, L, r)$*-simple* $(\mathsf{NIP}, \mathsf{SIG})$*-reduction* $r$*-*$\mathcal{R}$*, there exists an algorithm* $\mathcal{M}$ *that solves* NIP *such that*

$$\begin{aligned}\mathsf{Adv}^{\mathsf{NIP}}_{\mathcal{M}}(\lambda) &\geq L(\lambda, N, 1) - r \cdot (4\delta_{\mathcal{R}} + \varepsilon_{\mathsf{Sig}} + \varepsilon_{\mathrm{RO}}) - r/N,\\ \mathbf{T}(\mathcal{M}) &\leq r \cdot (N \cdot \mathbf{T}(\mathcal{R}) + N(N-1)\mathbf{T}(\mathsf{VerK}) + \mathbf{T}(\mathsf{Sig})),\end{aligned}$$

*where* $\mathbf{T}(X)$ *denotes the running time of* $X$*.*

The proof of Theorem 2 is almost identical to the proof of Theorem 1, and we can use the same proof technique as [BJLS16, Theorem 4]. Therefore it is omitted. Also, the interpretation of Theorem 2 is almost identical to the interpretation of Theorem 1. Since both the advantage and the running time are multiplied by $r$, $r$ is canceled when the reduction loss is calculated. As a result, the reduction loss from the multi-user security to an NIP is lower bounded by the number of users $N$ if $\varepsilon_{\mathsf{Sig}}, \varepsilon_{\mathrm{RO}}, \delta_{\mathcal{R}}$ are negligibly small, similarly to the interpretation of Theorem 1.

### 3.4 Discussion

Theorem 1 implies that to achieve tight security, at least one of the following conditions should be held.

(C1) SIG's security is based on interactive problems,

(C2) SIG is not signature statistically close, ($\varepsilon_{\mathsf{Sig}} \neq \mathsf{negl}$)

(C3) SIG is not RO statistically close, ($\varepsilon_{\mathrm{RO}} \neq \mathsf{negl}$)

(C4) The adversary's view given by a reduction $\mathcal{R}$ is statistically distinguishable from that in the real $N$-**MU-UF-S** ($\delta_{\mathcal{R}} \neq \mathsf{negl}$).

Table 2 summarizes which conditions existing tightly-secure signature schemes satisfy. Further,

- we do not want to rely on the hardness of interactive problems (unlike [WLG+19]),
- signatures signed by different (valid) secret keys should be indistinguishable. We note that, to reduce a forgery to solving a NIP without guessing corruption users, the reduction $\mathcal{R}$ must know (at least) one of valid secret keys in case of the corruption query, and $\mathcal{R}$ can use a forged signature to solve the NIP only if the forged one was generated by another valid secret key. If signatures signed by different keys were distinguishable, the adversary would learn information about what secret key $\mathcal{R}$ has from the signing query responses, and would always be able to output a forgery associated with that key, meaning that $\mathcal{R}$ would fail to solve the NIP. Therefore, if signatures signed by different keys are not statistically close, they should be computationally indistinguishable, meaning that a decisional assumption is needed (as in [GJ18]), and
- the adversary's view given by a reduction should be distinguishable from that in the real game. If they are not statistically close, they should be computationally indistinguishable, meaning that a decisional assumption is needed (as in [Bad14, BHJ+15, ABP19, DGJL21, PW22]).

From the above considerations, the only approach left to construct a tightly $N$-MU-UF-CMA-C secure signature from search assumptions is making $Q(\mathsf{sk}, \mathsf{m})$ and $Q(\mathsf{sk}', \mathsf{m})$ distinguishable, shown in the last row in Table 2.

Table 2: Conditions existing tightly secure schemes satisfy to avoid the impossibility results.

| Scheme | (C1) not NIP | (C2) $\varepsilon_{\mathsf{Sig}} \neq \mathsf{negl}$ | (C3) $\varepsilon_{\mathrm{RO}} \neq \mathsf{negl}$ | (C4) $\delta_{\mathcal{R}} \neq \mathsf{negl}$ |
|---|---|---|---|---|
| [WLG+19] | ✓ | - | - | - |
| [GJ18] | - | ✓ | - | - |
| [DGJL21, PW22] | - | - | ✓ | ✓ |
| [Bad14, BHJ+15, ABP19] | - | - | - | ✓ |
| Ours | - | - | ✓ | - |

# 4 New Construction

We provide our (failed) approach to construct the desired signature scheme. Our idea is the combination of the CDH-based 5-move identification scheme [KLP17] and the sequential-OR technique for multi-round interactive proofs, proposed in [FGQ+23]. The construction is as follows.

- $\mathsf{Setup}(1^\lambda)$: Output the description of a multiplicative group $\mathbb{G}$, its order $p$, its generator $g$, and the description of hash functions $\mathsf{H} : \{0,1\}^* \to \mathbb{G}$ and $\mathsf{H}' : \{0,1\}^* \to \mathbb{Z}_p$ as $\mathsf{par}$.
- $\mathsf{Gen}(\mathsf{par})$: Sample $\mathsf{sk}_0, \mathsf{sk}_1 \leftarrow\!\!\$\ \mathbb{Z}_p$, $b \leftarrow\!\!\$\ \{0,1\}$ and compute $\mathsf{pk}_0 = g^{\mathsf{sk}_0}, \mathsf{pk}_1 = g^{\mathsf{sk}_1}$. Output $\mathsf{sk} := (\mathsf{sk}_b, b), \mathsf{pk} := (\mathsf{pk}_0, \mathsf{pk}_1)$.
- $\mathsf{Sig}(\mathsf{sk}, \mathsf{m})$: Simulate a transcript of the 5-move ID protocol for $\mathsf{pk}_{1-b}$ with its simulation algorithm $\mathsf{Sim}$:

$$(R_{1-b}, h_{1-b}, R'_{1-b}, h'_{1-b}, s_{1-b}) \leftarrow \mathsf{Sim}(\mathsf{pk}_{1-b})$$

Then, compute a real transcript of the 5-move ID protocol for $\mathsf{pk}_b$ with its prover algorithm $P = (P_1, P_2, P_3)$:

$$\begin{aligned}
A_b &:= (a_b, a'_b) \leftarrow\!\!\$\ \mathbb{G} \times \mathbb{Z}_p, \\
(R_b, r) &\leftarrow P_1(\mathsf{sk}_b) = g^r \quad (r \leftarrow\!\!\$\ \mathbb{Z}_p) \\
a_{1-b} &:= h_{1-b} / \mathsf{H}(\mathsf{pk}_{1-b}, R_0, R_1, A_b, \mathsf{m}) \\
a'_{1-b} &:= h'_{1-b} - \mathsf{H}'(\mathsf{pk}_{1-b}, R_0, R_1, R'_{1-b}, A_b, \mathsf{m}) \\
A_{1-b} &:= (a_{1-b}, a'_{1-b}) \\
h_b &:= \mathsf{H}(\mathsf{pk}_b, R_0, R_1, A_{1-b}, \mathsf{m}) \times a_b \\
R'_b &\leftarrow P_2(\mathsf{sk}_b, R_b, h_b, r) = (R_{Lb} := h_b^{\mathsf{sk}_b}, R_{Rb} := h_b^r) \\
h'_b &:= \mathsf{H}'(\mathsf{pk}_b, R_0, R_1, R'_b, A_{1-b}, \mathsf{m}) + a'_b \\
s_b &\leftarrow P_3(\mathsf{sk}_b, R_b, h_b, R'_b, h'_b, r) = \mathsf{sk}_b \cdot h'_b + r.
\end{aligned}$$

Output $\sigma := (R_0, R'_0, R_1, R'_1, A_0, A_1, s_0, s_1)$.

- $\mathsf{Ver}(\mathsf{pk}, \mathsf{m}, \sigma = (R_0, R'_0, R_1, R'_1, A_0, A_1, s_0, s_1))$: Parse $A_0 = (a_0, a'_0), A_1 = (a_1, a'_1)$. For each $b \in \{0,1\}$, compute

$$\begin{aligned}
h_b &:= \mathsf{H}(\mathsf{pk}_b, R_0, R_1, A_{1-b}, \mathsf{m}) \times a_b, \\
h'_b &:= \mathsf{H}'(\mathsf{pk}_b, R_0, R_1, R'_b, A_{1-b}, \mathsf{m}) + a'_b, \\
v_b &\leftarrow V(\mathsf{pk}_b, R_b, R'_b, h_b, h'_b, s_b) \\
&= [R_b = g^{s_b} \mathsf{pk}_b^{-h'_b} \wedge R_{Rb} = h_b^{s_b} R_{Lb}^{-h'_b}].
\end{aligned}$$

If $v_0 = v_1 = 1$, output 1; otherwise, output 0.

The correctness of the scheme follows from the correctness of the identification scheme and the OR-proof technique. We now show its security.

**Theorem 3.** *Under the CDH assumption, the above scheme has $N$-MU-UF-CMA-C security with the reduction loss of $O(q_H)$ in the random oracle model, where $q_H$ is the number of $\mathsf{H}$ queries made by $\mathcal{A}$.*

*Proof.* Let $(i^*, \sigma^* = (R_0^*, {R'}_0^*, R_1^*, {R'}_1^*, A_0^*, A_1^*, s_0^*, s_1^*), \mathsf{m}^*)$ be $\mathcal{A}$'s output. Let

$$\begin{aligned} b^* &= b_{i^*} \\ H_b &= \mathsf{H}(\mathsf{pk}_{i^*,b}, R_0^*, R_1^*, A_{1-b}^*, \mathsf{m}^*) \\ H'_b &= \mathsf{H}'(\mathsf{pk}_{i^*,b}, R_0^*, R_1^*, {R'}_b^*, A_{1-b}^*, \mathsf{m}^*) \end{aligned}$$

for $b \in \{0,1\}$. In the following, we use the fact that, if $\sigma^*$ is accepted, there exists $(h_0, h'_0, h_1, h'_1)$ s.t.

$$\left.\begin{aligned} h_b &= H_b \times a_b^* \\ h'_b &= H'_b + {a'}_b^* \\ R_b^* &= g^{s_b^*} \mathsf{pk}_{i^*,b}^{-h'_b} \\ R_{Rb}^* &= h_b^{s_b^*} {R_{Lb}^*}^{-h'_b} \end{aligned}\right\} \tag{1}$$

for $b \in \{0,1\}$.

Game 0: Original attack game.

$$\epsilon_0 = \mathsf{Adv}^{N\text{-MU-UF-CMA-C}}_{\mathcal{A},\mathsf{SIG}}(\lambda)$$

Game 1: After the adversary outputs the forged signature, the challenger returns 0 if $H_{b^*}$ was queried before $H_{1-b^*}$. Since $\mathcal{A}$ has no information about $b^*$, we have

$$\epsilon_1 = (1/2)\epsilon_0$$

Game 2: After the adversary outputs the forged signature, the challenger returns 0 if $H'_{b^*}$ was queried before $H_{b^*}$. The next lemma shows

$$|\epsilon_2 - \epsilon_1| = 1/p.$$

**Lemma 5.** *If $H'_{b^*}$ was queried before $H_{b^*}$, $\sigma^*$ is rejected with probability $1 - 1/p$.*

*Proof.* We can assume that $H_{1-b^*}$ was queried before $H_{b^*}$. Further, we assume that $H'_{b^*}$ was queried before $H_{b^*}$, When $H_{b^*}$ is queried, $R_0^*, R_1^*, (a_1^*, {a'}_1^*)$, $(a_0^*, {a'}_0^*)$,
$(R_{Lb^*}^*, R_{Rb^*}^*)$ and $H'_{b^*}, H_{1-b^*}$ are all fixed. From these values, values of $h'_{b^*}, s_{b^*}^*, h_{b^*}$ that satisfy the following equations in Eq.(1) are uniquely determined.

$$\begin{aligned} h'_{b^*} &= H'_{b^*} + {a'}_{b^*}^*, \quad (h'_{b^*} \text{ is fixed}) \\ R_{b^*} &= g^{s_{b^*}^*} \mathsf{pk}_{i^*,b^*}^{-h'_{b^*}}, \quad (s_{b^*}^* \text{ is fixed}) \\ R_{Rb^*}^* &= h_{b^*}^{s_{b^*}^*} {R_{Lb^*}^*}^{-h'_{b^*}}. \quad (h_{b^*} \text{ is fixed}) \end{aligned}$$

Therefore, the equation

$$h_{b^*} = a_{b^*}^* \times H_{b^*}$$

in Eq.(1) is satisfied with the probability $1/p$, since $H_{b^*} = \mathsf{H}(\mathsf{pk}_{i^*,b^*}, R_0^*, R_1^*, A_{1-b^*}^*, \mathsf{m}^*)$ is randomly chosen. □

Reduction: Let $X = g^x, Y = g^y$ be an instance of the CDH problem. For $i = 1, 2, \ldots, N$, the reduction $\mathcal{R}$ chooses $b_i \leftarrow\!\$\ \{0,1\}$, generates $(\mathsf{pk}_{i,1-b_i}, \mathsf{sk}_{i,1-b_i})$ normally, and sets $\mathsf{pk}_{i,b_i} := Xg^{x_i}$ $(x_i \leftarrow\!\$\ \mathbb{Z}_p)$, $\mathsf{pk}_i := (\mathsf{pk}_{i,0}, \mathsf{pk}_{i,1})$. Since $\mathcal{R}$ knows $\mathsf{sk}_{i,1-b_i}$, it can generate a valid signature for any messages, and answer corrupt queries correctly. Further, $\mathsf{Sig}(\mathsf{sk}_{i,b_i}, \mathsf{m})$ and $\mathsf{Sig}(\mathsf{sk}_{i,1-b_i}, \mathsf{m})$ have the same probability distribution.

Then, $\mathcal{R}$ executes the adversary $\mathcal{A}$ on input $\{\mathsf{pk}_i\}_{i\in[N]}$ and answers oracle queries as follows:

- Simulation of $\mathsf{H}'$ oracle: When $(\mathsf{pk}, R_0, R_1, R', A, \mathsf{m})$ is queried, return randomly chosen $h' \leftarrow\!\$\, \mathbb{Z}_p$.
- Simulation of $\mathsf{H}$ oracle: When $(\mathsf{pk}, R_0, R_1, a, a', \mathsf{m})$ is queried, if $\mathsf{pk} = \mathsf{pk}_{i,b_i}$, $\mathcal{R}$ returns $H \leftarrow\!\$\, \mathbb{G}$ and adds $(\mathsf{pk}_{i,b_i}, R_0, R_1, a, a', \mathsf{m})$ to $L_1$. If $\mathsf{pk} = \mathsf{pk}_{i,1-b_i}$ and there exists $(\mathsf{pk}_{i,b_i}, R_0, R_1, a_{1-b_i}, a'_{1-b_i}, \mathsf{m}) \in L_1$ for some $(a_{1-b_i}, a'_{1-b_i})$, then (if there are multiple $a_{1-b_i}$, choose one randomly, and) $\mathcal{R}$ chooses $y_j \leftarrow\!\$\, \mathbb{Z}_p$ and returns $Yg^{y_j}/a_{1-b_i}$. Add $(\mathsf{pk}_{i,1-b_i}, R_0, R_1, a, a', \mathsf{m}, y_j)$ to $L_2$. Otherwise, returns $H \leftarrow\!\$\, \mathbb{G}$.
- $\mathtt{Corr}(i)$ query: Return $\mathsf{sk}_{i,1-b_i}$.
- $\mathtt{Sig}(i, \mathsf{m})$ query: Generate a signature by using $\mathsf{sk}_{i,1-b_i}$, and return the signature.

$\mathcal{A}$ outputs $(i^*, \mathsf{m}^*, \sigma^*)$, where

$$\sigma^* = (R_0^*, {R'_0}^*, R_1^*, {R'_1}^*, A_0^*, A_1^*, s_0^*, s_1^*).$$

If $H_{b^*}$ was queried before $H_{1-b^*}$ or $H'_{b^*}$ was queried before $H_{b^*}$ or $\mathsf{Ver}(\mathsf{pk}_{i^*}, \mathsf{m}^*, \sigma^*) = 0$, $\mathcal{R}$ outputs randomly chosen element $Z \leftarrow\!\$\, \mathbb{G}$.

Now consider the case that $H_{1-b^*}, H_{b^*}, H'_{b^*}$ are queried in this order, and $\mathsf{Ver}(\mathsf{pk}_{i^*}, \mathsf{m}^*, \sigma^*) = 1$.

When $H_{1-b^*} = \mathsf{H}(\mathsf{pk}_{i^*,1-b^*}, R_0^*, R_1^*, a_{b^*}^*, {a'}_{b^*}^*, \mathsf{m}^*)$ was queried, $(\mathsf{pk}_{i^*,1-b^*}, R_0^*, R_1^*, a_{b^*}^*, {a'}_{b^*}^*, \mathsf{m}^*)$ was added to $L_1$. Suppose that there is only one such entry. In this case, when $H_{b^*} = \mathsf{H}(\mathsf{pk}_{i^*,b^*}, R_0^*, R_1^*, a_{1-b^*}^*, {a'}_{1-b^*}^*, \mathsf{m}^*)$ was queried, $H_{b^*} = Yg^{y_j}/a_{b^*}^*$ was returned, and $(\mathsf{pk}_{i^*,b^*}, R_0^*, R_1^*, a_{1-b^*}^*, {a'}_{1-b^*}^*, \mathsf{m}^*, y_j)$ was added to $L_2$. $\mathcal{R}$ returns

$$Z := R_{Lb^*}^* / X^{y_j} Y^{x_{i^*}} g^{x_{i^*} y_j}.$$

Now define $\tilde{y}$ as

$$\tilde{y} := y + y_j.$$

Then,

$$h_{b^*} := H_{b^*} \times a_{b^*}^* = g^{\tilde{y}}.$$

**Lemma 6.** *If $H'_{b^*}$ is queried after $H_{1-b^*}$ and $H_{b^*}$, and the following equation does not hold, $\sigma^*$ is rejected with probability $1 - 1/p$.*

$$R_{Lb^*}^* = \mathsf{pk}_{i^*,b^*}^{\tilde{y}}. \tag{2}$$

*Proof.* When $H'_{b^*}$ is queried, $R_0^*, R_1^*, (a_1^*, {a'}_1^*), (a_0^*, {a'}_0^*)$, $(R_{Lb^*}^*, R_{Rb^*})$ and $H_0, H_1$ are all fixed. From the following equations

$$R_{b^*}^* = g^{s_{b^*}^*} \mathsf{pk}_{i^*,b^*}^{-h'_{b^*}},$$
$$R_{Rb^*}^* = h_{b^*}^{s_{b^*}^*} {R_{Lb^*}^*}^{-h'_{b^*}},$$

in Eq.(1), $\sigma^*$ is accepted only if

$$\begin{pmatrix} \log_g R_{b^*}^* \\ \log_g R_{Rb^*}^* \end{pmatrix} = \begin{pmatrix} 1 & -\log_g \mathsf{pk}_{i^*,b^*} \\ \tilde{y} & -\log_g R_{Lb^*}^* \end{pmatrix} \begin{pmatrix} s_{b^*}^* \\ h'_{b^*} \end{pmatrix}$$

holds. If Equation (2) does not hold, the matrix

$$\begin{pmatrix} 1 & -\log_g \mathsf{pk}_{i^*,b^*} \\ \tilde{y} & -\log_g R_{Lb^*}^* \end{pmatrix}$$

is regular, so there exists only one value of $h'_{b^*}$, and the probability the value satisfies $h'_{b^*} - a'_{b^*} = H'_{b^*}$ is $1/p$, since $H'_{b^*} = \mathsf{H}'(\mathsf{pk}_{i^*,b^*}, R_0^*, R_1^*, {R'}_b^*, A_{1-b}^*, \mathsf{m})$ is randomly chosen independently from other values. Therefore, the signature is rejected with probability $1 - 1/p$. □

If Equation (2) holds,

$$\begin{aligned} R^*_{L(b^*)} &= \mathsf{pk}^{\tilde{y}}_{i^*,b^*} \\ &= (Xg^{x_{i^*}})^{\tilde{y}} \\ &= (g^{x+x_{i^*}})^{y+y_j} = g^{(x+x_{i^*})(y+y_j)}. \end{aligned}$$

The $\mathcal{R}$'s output satisfies

$$Z = \frac{R^*_{Lb^*}}{X^{y_j} Y^{x_{i^*}} g^{x_{i^*} y_j}} = \frac{g^{(x+x_{i^*})(y+y_j)}}{X^{y_j} Y^{x_{i^*}} g^{x_{i^*} y_j}} = g^{xy}.$$

Therefore,

$$\begin{aligned} \mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) &= \Pr[\text{Equation (2) holds}] \\ &\geq \Pr[\text{Equation (2) holds} \wedge \sigma^* \text{ is accepted}] \\ &= \Pr[\sigma^* \text{ is accepted}] - \Pr[\sigma^* \text{ is accepted} \wedge \text{Equation (2) does not hold}] \\ &\geq \Pr[\sigma^* \text{ is accepted}] - 1/p, \\ \epsilon_2 &\leq \Pr[\sigma^* \text{ is accepted}] \\ &\leq \mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) + 1/p. \end{aligned}$$

Consequently, we have

$$\mathsf{Adv}^{N\text{-MU-UF-CMA-C}}_{\mathcal{A},\mathsf{SIG}}(\lambda) \leq 2(\mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) + 2/p).$$

If there are $q_H$ entries: In this case, we have to estimate the success probability as

$$\mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) = \frac{1}{q_H} \Pr[\text{Equation (2) holds}].$$

Thus we have

$$\begin{aligned} \mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) &\geq \frac{1}{q_H}(\Pr[\sigma^* \text{ is accepted}] - 1/p), \\ \epsilon_2 &\leq q_H \mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) + 1/p \end{aligned}$$

and

$$\mathsf{Adv}^{N\text{-MU-UF-CMA-C}}_{\mathcal{A},\mathsf{SIG}}(\lambda) \leq 2(q_H \mathsf{Adv}^{CDH}_{\mathcal{R}}(\lambda) + 2/p).$$

□

# 5 Conclusion

In this work, we tried to construct a signature scheme whose *multi-user security with corruption* can be *tightly* reduced to *search assumptions*. We first revealed the new conditions that the highest secure signature schemes must satisfy. This result suggests that constructions based on the OR-proof are promising. Second, by combining the 5-move CDH-based identification scheme [KLP17] and the OR-Proof technique for multi-round interactive protocols [FGQ+23], we constructed a new signature scheme. As a result, we made the reduction loss from its multi-user security with corruption to the CDH assumption independent of the number of users. However, this approach failed as its loss depended on the number of queries to the RO. The existence of a signature scheme whose multi-user security with corruption is tightly reduced to search assumptions remains still open.

## Acknowledgments

The authors thank the anonymous reviewers of CFAIL 2024 for their constructive comments and suggestions. Keitaro Hashimoto was partially supported by JST CREST JPMJCR22M1, Japan.

# References

[ABP19] Michel Abdalla, Fabrice Benhamouda, and David Pointcheval. On the tightness of forward-secure signature reductions. *Journal of Cryptology*, 32(1):84–150, January 2019.

[AGO11] Masayuki Abe, Jens Groth, and Miyako Ohkubo. Separating short structure-preserving signatures from non-interactive assumptions. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 628–646. Springer, Heidelberg, December 2011.

[Bad14] Christoph Bader. Efficient signatures with tight real world security in the random-oracle model. In Dimitris Gritzalis, Aggelos Kiayias, and Ioannis G. Askoxylakis, editors, *CANS 14*, volume 8813 of *LNCS*, pages 370–383. Springer, Heidelberg, October 2014.

[BHJ+15] Christoph Bader, Dennis Hofheinz, Tibor Jager, Eike Kiltz, and Yong Li. Tightly-secure authenticated key exchange. In Yevgeniy Dodis and Jesper Buus Nielsen, editors, *TCC 2015, Part I*, volume 9014 of *LNCS*, pages 629–658. Springer, Heidelberg, March 2015.

[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, Part II*, volume 9666 of *LNCS*, pages 273–304. Springer, Heidelberg, May 2016.

[Che05] Benoît Chevallier-Mames. An efficient CDH-based signature scheme with a tight security reduction. In Victor Shoup, editor, *CRYPTO 2005*, volume 3621 of *LNCS*, pages 511–526. Springer, Heidelberg, August 2005.

[DGJL21] Denis Diemert, Kai Gellert, Tibor Jager, and Lin Lyu. More efficient digital signatures with tight multi-user security. In Juan Garay, editor, *PKC 2021, Part II*, volume 12711 of *LNCS*, pages 1–31. Springer, Heidelberg, May 2021.

[FGQ+23] Pierre-Alain Fouque, Adela Georgescu, Chen Qian, Adeline Roux-Langlois, and Weiqiang Wen. A generic transform from multi-round interactive proof to NIZK. In Alexandra Boldyreva and Vladimir Kolesnikov, editors, *PKC 2023, Part II*, volume 13941 of *LNCS*, pages 461–481. Springer, Heidelberg, May 2023.

[GJ03] Eu-Jin Goh and Stanislaw Jarecki. A signature scheme as secure as the Diffie-Hellman problem. In Eli Biham, editor, *EUROCRYPT 2003*, volume 2656 of *LNCS*, pages 401–415. Springer, Heidelberg, May 2003.

[GJ18] Kristian Gjøsteen and Tibor Jager. Practical and tightly-secure digital signatures and authenticated key exchange. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 95–125. Springer, Heidelberg, August 2018.

[KLP17] Eike Kiltz, Julian Loss, and Jiaxin Pan. Tightly-secure signatures from five-move identification protocols. In Tsuyoshi Takagi and Thomas Peyrin, editors, *ASIACRYPT 2017, Part III*, volume 10626 of *LNCS*, pages 68–94. Springer, Heidelberg, December 2017.

[PR20] Jiaxin Pan and Magnus Ringerud. Signatures with tight multi-user security from search assumptions. In Liqun Chen, Ninghui Li, Kaitai Liang, and Steve A. Schneider, editors, *ESORICS 2020, Part II*, volume 12309 of *LNCS*, pages 485–504. Springer, Heidelberg, September 2020.

[PW22] Jiaxin Pan and Benedikt Wagner. Lattice-based signatures with tight adaptive corruptions and more. In Goichiro Hanaoka, Junji Shikata, and Yohei Watanabe, editors, *PKC 2022, Part II*, volume 13178 of *LNCS*, pages 347–378. Springer, Heidelberg, March 2022.

[WLG+19] Ge Wu, Jian-Chang Lai, Fu-Chun Guo, Willy Susilo, and Fu-Tai Zhang. Tightly secure public-key cryptographic schemes from one-more assumptions. *Journal of Computer Science and Technology*, 34:1366–1379, 2019.