Multi-Dimensional Sub/Super-Range Signatures

Masahito Ishizaka and Shinsaku Kiyomoto

KDDI Research, Inc., Saitama, Japan.
{ma-ishizaka,kiyomoto}@kddi-research.jp

Abstract. In time-specific signatures (TSS) [Paterson & Quaglia, SCN'10] [Ishizaka & Kiyomoto, ISC'20] with T numerical values, each signer is given a secret-key associated with a numerical value $t \in [0, T-1]$ and each signature on a message is generated under a numerical range $[L, R]$ s.t. $0 \le L \le R \le T-1$. A signer with t can correctly generate a signature under $[L, R]$ if t is truly included in $[L, R]$, i.e., $t \in [L, R]$.
As a generalized primitive of TSS, we propose multi-dimensional *sub*-range signatures (MDSBRS). As a related primitive, we also propose multi-dimensional *super*-range signatures (MDSPRS). In MDSBRS (resp. MDSPRS) with $D \in \mathbb{N}$ dimensions, each secret-key is associated with a set of D ranges $\{[l_i, r_i] \mid i \in [1, D]\}$ s.t. $0 \le l_i \le r_i \le T_i - 1$ and a threshold value $d \in [1, D]$, and it correctly produces a signature on any message under a set of D ranges $\{[L_i, R_i] \mid i \in [1, D]\}$ s.t. $0 \le L_i \le R_i \le T_i - 1$, if and only if total number of key-ranges every one $[l_i, r_i]$ of which is a *sub*-range (resp. *super*-range) of the corresponded signature-range $[L_i, R_i]$, i.e., $L_i \le l_i \le r_i \le R_i$ (resp. $l_i \le L_i \le R_i \le r_i$), is more than $d-1$. We show that, by extending (or generalizing) an existing TSS scheme, we obtain MDSBRS and MDSPRS schemes each one of which is secure, i.e., existentially unforgeable and perfectly (signer-)private, under standard assumption and asymptotically efficient.

Keywords: Multi-dimensional sub-range signatures, Multi-dimensional super-range signatures, Existential unforgeability, Perfect (signer-)privacy.

1 Introduction

Time-Specific Encryption. Paterson and Quaglia [10] proposed time-specific encryption (TSE) (or interchangeably range encryption (RE)). In the encryption system, we assume that there exists a trusted key-generation authority which generates a master public-key *mpk* and a master secret-key *msk* from an integer $T \in \mathbb{N}$ denoting total number of *time-periods* (or *numerical values*). The trusted authority generates a secret-key for a time-period $t \in [0, T-1]$ by using the master secret-key. The secret-key is securely sent to a user. A ciphertext C of a plaintext m is associated with a range $[L, R]$ s.t. $0 \le L \le R \le T-1$. Any user with a secret-key for t s.t. $t \in [L, R]$ can correctly decrypt the ciphertext. It has been clarified by [10,6,7,4] that TSE schemes with different characteristics in efficiency can be (generically) constructed from identity-based encryption

(IBE), broadcast encryption, hierarchical IBE (HIBE), forward-secure encryption (FSE) and wildcarded IBE (WIBE). For instance, we can obtain a concrete TSE scheme whose efficiency $(|mpk|, |sk|, |C|)$ is informally $(\mathcal{O}(\log T), \mathcal{O}(\log T), \mathcal{O}(\log T))$ (resp. $(\mathcal{O}(\log T), \mathcal{O}(1), \mathcal{O}(\log^2 T))$) from the generic TSE construction based on an IBE (resp. WIBE) scheme proposed by Paterson and Quaglia [10] (resp. Ishizaka and Kiyomoto [4]). Moreover, the TSE scheme based on an FSE scheme proposed by Kasamatsu et al. [6] achieves $(|mpk|, |sk|, |C|) = (\mathcal{O}(\log T), \mathcal{O}(\log^2 T), \mathcal{O}(1))$.

Time-Specific Signatures. In the original work of TSE [10], the authors mentioned a primitive called time-specific signatures (TSS) as the digital signature analogue of TSE and posed finding a secure TSS construction as an open problem.

Ishizaka and Kiyomoto [5] formally defined its syntax and security requirements, namely *existential unforgeability* (which informally guarantees that any multiple colluding users every one of which does not have a secret-key associated with a numerical value t s.t. $t \in [L, R]$ for a range $[L, R]$ cannot forge a correct signature under the range) and *perfect (signer-)privacy* (which guarantees that any verifier given a signature cannot get any specific information about the numerical value of the signer). They proposed the first two asymptotically-efficient concrete schemes secure under standard assumptions. The first (resp. second) one is based on forward-secure signatures (FSS) (resp. wildcarded identity-based ring signatures), whose efficiency $(|mpk|, |sk|, |\sigma|)$ is $(\mathcal{O}(\log T), \mathcal{O}(\log T), \mathcal{O}(\log T))$ (resp. $(\mathcal{O}(\log T), \mathcal{O}(1), \mathcal{O}(\log^2 T))$).

As a direct application of TSS, *numerical-range based anonymous questionnaire* was introduced in [5]. Each user is associated with a numerical value $t \in [0, T-1]$ and given a secret-key for the value. The user can anonymously fill in a questionnaire (or sign a message) with declaring a range $[L, R]$ s.t. $t \in [L, R]$. The standard TSS can deal with only one-dimensional numerical value. It is obvious that *multi-dimensional* TSS are more practically-desirable. Finding a secure multi-dimensional TSS has been posed as an open problem in [5].

Multi-Dimensional Sub-Range Signatures. As a generalization of TSS, we propose a primitive named multi-dimensional sub-range signatures (MDSBRS). It is parameterized by integers D and $\{T_i \mid i \in [1, D]\}$. Each secret-key is associated with a set of ranges $\{[l_i, r_i] \subseteq [0, T_i - 1] \mid i \in [1, D]\}$. Such a secret-key can produce a signature on any message under a set of ranges $\{[L_i, R_i] \subseteq [0, T_i - 1] \mid i \in [1, D]\}$, if (and only if) every key-range $[l_i, r_i]$ is a *sub*-range (or *sub*set) of the corresponded signature-range $[L_i, R_i]$, i.e., $[l_i, r_i] \subseteq [L_i, R_i]$. Note that if we set $D = 1$ and restrict the key-range $[l, r]$ to one satisfying $l = r (=: t)$ for a numerical value t, the primitive is TSS itself. For instance, one-dimensional MDSBRS are illustrated in Fig. 1. There are two key-ranges with ✓, i.e., $[6, 11]$ and $[16, 16]$, and two key-ranges with ×, i.e., $[0, 1]$ and $[21, 26]$. Under a signature-range $[L, R] = [4, 23]$, both of the key-ranges with ✓ can correctly sign, and either of the key-ranges with × cannot correctly sign. 2-dimensional MDSBRS are illustrated in Fig. 2. Under a two-dimensional signature-range $\{[L_1, R_1],$

$[L_2, R_2]\} = \{[1,5],[2,12]\}$, both of the two two-dimensional key-ranges with ✓ can correctly sign, and either of the four two-dimensional key-ranges with × or $*$ cannot correctly sign.

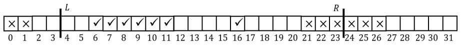

Fig. 1. An illustration of MDSBRS with $D = 1$ and $T = 32$.

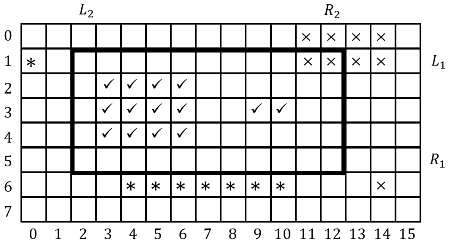

Fig. 2. An illustration of MDSBRS with $D = 2$, $T_1 = 8$ and $T_2 = 16$.

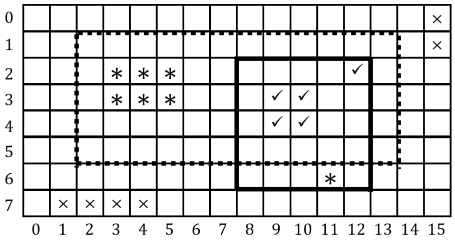

Fig. 3. An illustration of MDSBRS with $D = 2$, $T_1 = 8$ and $T_2 = 16$ for explaining *key-delegatability*.

Although the current MDSBRS have already been more generalized than the naive multi-dimensional TSS, we would like to further generalize the primitive by adding the following 2 properties[1].

- *Key-delegatability*: There exists an efficiently-computable one-way key-delegation algorithm which transforms a secret-key for a set of key-ranges $\{[l_i, r_i] \mid i \in [1, D]\}$ into one for any *weaker* set of key-ranges $\{[l'_i, r'_i] \mid i \in [1, D]\}$ s.t. every key-range satisfies $0 \leq l'_i \leq l_i \leq r_i \leq r'_i \leq T_i - 1$. In an example of two-dimensional MDSBRS in Fig. 3, either of the two (two-dimensional) key-ranges with ✓ can be a correct key-delegator for both of the key-ranges edged by a bold or dashed line. Either of the two key-ranges with $*$ can be a correct key-delegator for any one of the edged key-ranges, but not both. Either of the two key-ranges with × cannot be a correct key-delegator for any one of the edged key-ranges.
- *d-out-of-D threshold signability*: Each secret-key is associated with not only a set of key-ranges $\{[l_i, r_i] \mid i \in [1, D]\}$, but also a threshold $d \in [1, D]$. Each signer successfully signs if (and only if) total number of key-ranges $\{[l_i, r_i]\}$ each of which is truly included in the corresponded signature-range $[L_i, R_i]$

[1] Encryption primitive analogue of MDSBRS (w/o *key-delegatability* and w/ a restriction s.t. $\bigwedge_{i=1}^{D} T_i = T$) has been proposed as *K-out-of-L (ciphertext-policy) multi-dimensional range encryption (MDRE)* in [8]. Since their MDRE scheme is technically common with our MDSBRS scheme in adopting a methodology based on *forward-secure* cryptosystem, it is possible that their scheme also implicitly has the key-delegatability. However, we do not verify that, because that is out of our scope.

is more than $d-1$. For the instance in Fig. 2, if $d = 1$, both of the two key-ranges with $*$ can correctly sign under the two-dimensional signature-range $\{[L_1, R_1], [L_2, R_2]\}$. The threshold d can be arbitrarily chosen at the secret-key generation. The threshold can be fixed for all secret-keys if that is practically more desirable.

Multi-Dimensional Super-Range Signatures. As a primitive closely related to MDSBRS, we propose multi-dimensional *super*-range signatures (MDSPRS). Formally, in MDSPRS, each secret-key is associated with a set of ranges $\{[L_i, R_i] \subseteq [0, T_i - 1] \mid i \in [1, D]\}$ and a threshold $d \in [1, D]$, and such a secret-key successfully produces a signature on any message under a set of ranges $\{[l_i, r_i] \subseteq [0, T_i - 1] \mid i \in [1, D]\}$, if (and only if) total number of key-ranges $\{[L_i, R_i]\}$ each of which is a *super*-range (or *super*set) of the corresponded signature-range $[l_i, r_i]$ is more than $d-1$, i.e., $\sum_{i \in [1,D] \text{ s.t. } [l_i, r_i] \subseteq [L_i, R_i]} 1 \geq d$. Its key-delegatability guarantees that there exists an algorithm which transforms a secret-key for key-ranges $\{[L_i, R_i] \mid i \in [1, D]\}$ into one for any *weaker* set of key-ranges $\{[L'_i, R'_i] \mid i \in [1, D]\}$ such that every key-range satisfies $0 \leq L_i \leq L'_i \leq R'_i \leq R_i \leq T_i - 1$.

Our Results. As the security requirements whom MDSBRS and MDSPRS schemes satisfy, we formally define existential unforgeability and perfect privacy. By extending (or generalizing) the TSS scheme based on FSS proposed in [5], we firstly propose a concrete MDSBRS scheme. We formally prove that the scheme is existentially unforgeable under the co-CDH assumption [3] and perfectly private. We show that technique behind the MDSBRS scheme (or the original TSS) effectively works for a construction of a secure MDSPRS scheme. To the best of our knowledge, our schemes are the first currently-known polylogarithmically-efficient ones.

Our Approach. We generalize a TSS scheme in [5] based on FSS to obtain an MDSBRS scheme.

Kasamatsu et al. [6,7] and Ishizaka and Kiyomoto [5] showed that TSE and TSS are functionally close to FSE and FSS, respectively. For instance, a TSS scheme in [5] is based on the following intuitive idea to construct a TSS scheme from an FSS scheme and a *backward-secure signatures (BSS)* scheme. A secret-key for $t \in [0, T-1]$ consists of a secret-key for t w.r.t. the FSS scheme and one for t w.r.t. the BSS scheme. A signature under $[L, R]$ s.t. $L \leq t \leq R$ consists of a signature under R w.r.t. the FSS scheme generated from the FSS secret-key for t (note: if $t \leq R$, the generation succeeds) and a signature under L w.r.t. the BSS scheme generated from the BSS secret-key for t (note: if $t \geq L$, the generation succeeds). Although the generic approach is ideally simple, it cannot achieve security against *colluding (signature-forging) attacks*[2]. [5] showed that the generic approach effectively works on a concrete FSS scheme, i.e., one obtained by applying the Canetti-Halevi-Katz transformation [2] (which converts

[2] If a user with $t < L$ colludes with another user with $t > R$, they can forge a correct signature under $[L, R]$.

a hierarchical identity-based cryptographic scheme into a forward-secure one) to a hierarchical identity-based signatures (HIBS) scheme by Chutterjee and Sarker [3]. The technique behind the TSS scheme to achieve security against the colluding attack is *tying up* the FSS and BSS secret-keys for t. Specifically, when we generate a secret-key for t, we firstly divide the *true* master secret-key (MSK) into two *pseudo* MSKs in an algebraic manner based on 2-out-of-2 Shamir's threshold secret sharing (SSS) [13]. Then, we generate the FSS and BSS secret-keys (for t) by using the two pseudo MSKs, respectively. When we generate a signature under $[L, R]$, we evolve the secret-keys into ones for the time periods R and L, respectively, then combine them to obtain a correct secret-key for $[L, R]$ which is based on the *true* MSK. By using the secret-key, we can generate a signature on any message.

We extend (or generalize) their approach in the following three steps.

Firstly, generalizing a point (or time period) t into a range $[l, r]$ is simple. A secret-key for $[l, r]$ consists of a secret-key for r w.r.t. the FSS scheme and one for l w.r.t. the BSS scheme. When we generate a signature under a range $[L, R]$, we firstly evolve the secret-keys into one for R and L, respectively (note: this succeeds, if $L \leq l \leq r \leq R$). Then, we combine the (two) secret-keys to obtain a correct secret-key for $[L, R]$. By using it, we generate a signature on a message.

Secondly, the TSS scheme in [5] has been already key-delegatable. Precisely, the FSS secret-key for r and the BSS secret-key for l can be evolved into an FSS one for $r' \geq r$ and a BSS one for $l' \leq l$, respectively.

Thirdly, for the multi-dimensionalization with d-out-of-D threshold signability, we adopt the technique behind a fuzzy IBE scheme [11] and a (ciphertext-policy) MDRE scheme [8]. Specifically, we use d-out-of-D SSS. A secret-key for key-ranges $\{[l_i, r_i] \mid i \in [1, D]\}$ is generated as follows. We firstly divide the true MSK in an algebraic manner based on the d-out-of-D SSS method into D pseudo *first-level* MSKs. For each dimension $i \in [1, D]$, we further divide the pseudo first-level MSK based on the 2-out-of-2 SSS method into two pseudo *second-level* MSKs. We finally generate an FSS secret-key for r_i and a BSS one for l_i from the two pseudo second-level MSKs, respectively. We generate a signature under signature-ranges $\{[L_i, R_i] \mid i \in [1, D]\}$ as follows. For every $i \in [1, D]$ s.t. $L_i \leq l_i \leq r_i \leq R_i$, we evolve the FSS secret-key for r_i and the BSS secret-key for l_i into ones for R_i and L_i, respectively. By properly combining them based on the secret-recovering algorithm w.r.t. SSS, if (and only if) total number of indices $i \in [1, D]$ s.t. $[l_i, r_i] \subseteq [L_i, R_i]$ is larger than $d - 1$, we obtain a correct secret-key for the signature-ranges $\{[L_i, R_i] \mid i \in [1, D]\}$ which is based on the true original MSK. By using it, we can correctly generate a signature under the signature-ranges on any message.

We remind us that MDSPRS consider whether a relation that a key-range $[L_i, R_i]$ is a super-range of the corresponded signature-range $[l_i, r_i]$ holds or not. The approach for MDSBRS effectively works to MDSPRS. We generate an FSS secret-key for L_i and a BSS secret-key for R_i. Note that if $L_i \leq l_i \leq r_i \leq R_i$, we depart from L_i (resp. R_i), move forward (resp. backward), then arrive at l_i (resp. r_i).

Paper Organization. Sect. 2 is a section for preliminaries. In Sect. 3, we provide syntax and security requirements of MDSBRS. We propose an MDSBRS scheme, then prove its security. In Sect. 4, we formally define MDSPRS, then prpose a secure scheme based on the same teachnique as the MDSBRS scheme. In Sect. 5, we analyse their efficiency and conclude the paper.

2 Preliminaries

Notations. For $\lambda \in \mathbb{N}$, 1^λ denotes a security parameter. $\mathbb{PPT}_\lambda$ denotes a set of all probabilistic algorithms running in time polynomial in λ. We say a function $f : \mathbb{N} \to \mathbb{R}$ is negligible if $\forall c \in \mathbb{N}$, $\exists x_0 \in \mathbb{N}$ s.t. $\forall x \geq x_0$, $f(x) \leq x^{-c}$. $\mathbb{NGL}_\lambda$ denotes a set of all functions negligible in λ. For $x \in \{0,1\}^n$ and $i \in [0, n-1]$, $x[i] \in \{0,1\}$ is its i-th bit.

Asymmetric Bilinear Groups of Prime Order. Let $\mathcal{G}_{BG}$ denote a probabilistic algorithm which generates bilinear groups of prime order. Let $\lambda \in \mathbb{N}$. Specifically, it takes 1^λ and randomly generates $(p, \mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T, e, g, \tilde{g})$, where p is a prime with bit length λ, $(\mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T)$ are multiplicative groups of order p, $(g, \tilde{g})$ are generators of $\mathbb{G}$ and $\tilde{\mathbb{G}}$, respectively, and $e : \mathbb{G} \times \tilde{\mathbb{G}} \to \mathbb{G}_T$ is a function which is computable in polynomial time and satisfies: (1) Bilinearity: for every $a, b \in \mathbb{Z}_p$, $e(g^a, \tilde{g}^b) = e(g, \tilde{g})^{ab}$, and (2) Non-degeneracy: $e(g, \tilde{g}) \neq 1_{\mathbb{G}_T}$, where $1_{\mathbb{G}_T}$ denotes the unit element of $\mathbb{G}_T$.

Hardness Assumption. Let $\mathcal{G}$ denote a probabilistic algorithm which takes 1^λ, where $\lambda \in \mathbb{N}$, then randomly generates $(p, \mathbb{G}, \tilde{\mathbb{G}}, g, \tilde{g})$, where p is a prime of bit length λ, $\mathbb{G}$ and $\tilde{\mathbb{G}}$ are multiplicative groups of order p, and g (resp. $\tilde{g}$) is a generator of $\mathbb{G}$ (resp. $\tilde{G}$).

Definition 1. *Co-Computational Diffie-Hellman (Co-CDH) assumption holds if* $\forall \lambda \in \mathbb{N}$, $\forall \mathcal{A} \in \mathbb{PPT}_\lambda$, $\exists \epsilon \in \mathbb{NGL}_\lambda$ *s.t.* $\mathtt{Adv}_{\mathcal{A}}^{\mathtt{Co\text{-}CDH}}(\lambda) := \Pr[g^{\alpha\beta} \leftarrow \mathcal{A}(p, \mathbb{G}, \tilde{\mathbb{G}}, g, \tilde{g}, g^\alpha, g^\beta, \tilde{g}^\beta)] < \epsilon$, *where* $(p, \mathbb{G}, \tilde{\mathbb{G}}, g, \tilde{g}) \leftarrow \mathcal{G}(1^\lambda)$ *and* $\alpha, \beta \overset{\mathrm{U}}{\leftarrow} \mathbb{Z}_p$.

Shamir's Threshold Secret Sharing [13]. In d-out-of-D threshold secret sharing, a secret is divided into D shares. If we collect more than $d-1$ shares among the D shares, we can identify the secret. Otherwise, we cannot get any information about the secret. Shamir [13] proposed a method based on polynomial interpolation. Let $\mathtt{GF}(p)$ denote a finite field of cardinality $p \geq D$. Let $s \in \mathtt{GF}(p)$ denote the secret. We define a $(d-1)$-dimensional polynomial $f : \mathtt{GF}(p) \to \mathtt{GF}(p)$ in a form of $f(x) = s + \sum_{i=1}^{d-1} A_i x^i$, where $A_1, \cdots, A_{d-1}$ are randomly chosen from $\mathtt{GF}(p)$. For each $i \in \{1, \cdots, D\}$, the i-th share is $f(\alpha_i)$, where α_i satisfies $\alpha_i \in \mathtt{GF}(p) \setminus \{0\} \setminus \cup_{j \in [1,D] \setminus \{i\}} \{\alpha_j\}$. A case where we obtain d shares amont the D shares means that we obtain a set of d equations with d unknown variables of $A_1, \cdots, A_{d-1}$ and s. Obviously, we can uniquely identify each of the unknown variables. For any $\mathbb{I} \subseteq \{1, \cdots, D\}$ with $|\mathbb{I}| \geq d$, if we get the shares $\{f(\alpha_i) \mid i \in \mathbb{I}\}$, we can identify $s = f(0)$ based on the following equation: $f(x) = \sum_{i \in \mathbb{I}} \Delta_{i,\mathbb{I}}(x) f(\alpha_i)$, where the Lagrange coefficient $\Delta_{i,\mathbb{I}}(x)$ is defined as $\prod_{j \in \mathbb{I} \setminus \{i\}} (x - \alpha_j)/(\alpha_i - \alpha_j)$.

3 Multi-Dimensional *Sub*-Range Signatures (MDSBRS)

We firstly formally define syntax and security requirements. We require every scheme to be existentially unforgeable and perfectly private. The definitions are naturally extended from the ones for TSS in [5]. We informally explain the technique behind our scheme in Subsect. 3.1 before formally describing the scheme in Subsect. 3.2. Our scheme is proven to be existentially unforgeable in Subsect. 3.3 and perfectly private in Subsect. 3.4.

Syntax. MDSBRS[3] (with $D \in \mathbb{N}$ dimensions) consist of the following 5 polynomial time algorithms, where $\texttt{Ver}$ is deterministic and the others are probabilistic.

Setup $\texttt{Setup}$: For $i \in [1, D]$, $T_i \in \mathbb{N}$ denotes total number of numerical values for the i-th dimension. The algorithm takes 1^λ, D and $\{T_i \mid i \in [1, D]\}$ as input, then outputs a master public-key mpk and a master secret-key msk. Concisely, we write $(mpk, msk) \leftarrow \texttt{Setup}(1^\lambda, D, \{T_i \mid i \in [1, D]\})$. Note that the other four algorithms implicitly take mpk as input.

Key-generation $\texttt{KGen}$: It takes msk, key-ranges $\{l_i, r_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i \in [1,D]} 0 \le l_i \le r_i \le T_i - 1$ and a threshold $d \in [1, D]$, then outputs a secret-key sk for the key-ranges. We write $sk \leftarrow \texttt{KGen}(msk, \{l_i, r_i \mid i \in [1, D]\}, d)$.

Key-delegation $\texttt{KDel}$: It takes sk for $\{l_i, r_i \mid i \in [1, D]\}$ and $d \in [1, D]$, extended key-ranges $\{l'_i, r'_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i \in [1,D]} 0 \le l'_i \le l_i \le r_i \le r'_i \le T_i - 1$, then outputs sk' for the extended key-ranges. We write $sk' \leftarrow \texttt{KDel}(sk, \{l'_i, r'_i \mid i \in [1, D]\})$.

Signing $\texttt{Sig}$: It takes sk for $\{l_i, r_i \mid i \in [1, D]\}$ and $d \in [1, D]$, a message $m \in \{0, 1\}^*$, and signature-ranges $\{L_i, R_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i \in [1,D]} 0 \le L_i \le R_i \le T_i - 1$, then outputs a signature σ. We write $\sigma \leftarrow \texttt{Sig}(sk, m, \{L_i, R_i \mid i \in [1, D]\})$.

Verification $\texttt{Ver}$: It takes σ, $m \in \{0, 1\}^*$ and $\{L_i, R_i \mid i \in [1, D]\}$, then outputs a bit 1/0. We write $1/0 \leftarrow \texttt{Ver}(\sigma, m, \{L_i, R_i \mid i \in [1, D]\})$.

We require every MDSBRS scheme to be correct. An MDSBRS scheme $\Sigma_{\text{SB}} = \{\texttt{Setup}, \texttt{KGen}, \texttt{KDel}, \texttt{Sig}, \texttt{Ver}\}$ is correct, if $\forall \lambda \in \mathbb{N}$, $\forall D \in \mathbb{N}$, $\forall T_1 \in \mathbb{N}, \cdots,$ $\forall T_D \in \mathbb{N}$, $\forall (mpk, msk) \leftarrow \texttt{Setup}(1^\lambda, D, \{T_i \mid i \in \mathbb{N}\})$, $\forall \{l_i, l'_i, r_i, r'_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i \in [1,D]} 0 \le l'_i \le l_i \le r_i \le r'_i \le T_i - 1$, $\forall d \in [1, D]$, $\forall sk \leftarrow \texttt{KGen}(msk, \{l_i, r_i \mid i \in [1, D]\}, d)$, $\forall sk' \leftarrow \texttt{KDel}(sk, \{l'_i, r'_i \mid i \in [1, D]\})$, $\forall m \in \{0, 1\}^*$, $\forall \{L_i, R_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i \in [1,D]} 0 \le L_i \le R_i \le T_i - 1 \bigwedge \sum_{i \in [1,D] \text{ s.t. } L_i \le l'_i \le r'_i \le R_i} 1 \ge d$, $\forall \sigma \leftarrow \texttt{Sig}(sk', m, \{L_i, R_i \mid i \in [1, D]\})$, $1 \leftarrow \texttt{Ver}(\sigma, m, \{L_i, R_i \mid i \in [1, D]\})$.

Security. As security, we consider *(adaptive) existential unforgeability* and *perfect privacy*. For a probabilistic algorithm $\mathcal{A}$, we consider an experiment for (adaptive) existential unforgeability in Fig. 4. For a probabilistic algorithm $\mathcal{A}$, we consider two experiments for perfect privacy in Fig. 4. The commands with a gray background (▨) are considered only in the experiment with $\beta = 1$.

[3] MDSBRS satisfying all of the following conditions are identical to the TSS considered in [5]: (i) $D = 1$. (ii) Lacking $\texttt{KDel}$ algorithm. (iii) Every range $[l, r]$ associated with a secret-key is a numerical value, i.e., $l = r (=: t)$.

Definition 2 ([9,12,5]). *An MDSBRS scheme Σ_{SB} is existentially unforgeable, if $\forall \lambda \in \mathbb{N}$, $\forall D \in \mathbb{N}$, $\forall T_1 \in \mathbb{N}, \cdots, \forall T_D \in \mathbb{N}$, $\forall \mathcal{A} \in \mathbb{PPT}_\lambda$, $\exists \epsilon \in \mathbb{NGL}_\lambda$ s.t. $\boldsymbol{Adv}^{\mathit{EUF\text{-}CMA}}_{\Sigma_{\mathrm{SB}},\mathcal{A},D,\{T_i|i\in[1,D]\}}(\lambda) := \Pr[1 \leftarrow \boldsymbol{Expt}^{\mathit{EUF\text{-}CMA}}_{\Sigma_{\mathrm{SB}},\mathcal{A}}(1^\lambda, D, \{T_i \mid i \in [1,D]\})] < \epsilon$.*

Definition 3 ([1,5]). *An MDSBRS scheme Σ_{SB} is perfectly private, if for every $\lambda \in \mathbb{N}$, every $D \in \mathbb{N}$, every $T_1 \in \mathbb{N}, \cdots$, every $T_D \in \mathbb{N}$ and every probabilistic algorithm $\mathcal{A}$, there exist probabilistic polynomial time algorithms $\{\widehat{\mathtt{Setup}}, \widehat{\mathtt{KGen}}, \widehat{\mathtt{KDel}}, \widehat{\mathtt{Sig}}\}$ such that $\boldsymbol{Adv}^{\mathit{PP}}_{\Sigma_{\mathrm{SB}},\mathcal{A},D,\{T_i|i\in[1,D]\}}(\lambda) := |\Pr[1 \leftarrow \boldsymbol{Expt}^{\mathit{PP}}_{\Sigma_{\mathrm{SB}},\mathcal{A},0}(1^\lambda, D, \{T_i \mid i \in [1,D]\})] - \Pr[1 \leftarrow \boldsymbol{Expt}^{\mathit{PP}}_{\Sigma_{\mathrm{SB}},\mathcal{A},1}(1^\lambda, D, \{T_i \mid i \in [1,D]\})]| = 0$.*

```
$\boldsymbol{Expt}^{\mathtt{EUF\text{-}CMA}}_{\Sigma_{\mathrm{SB}},\mathcal{A}}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$:
  $(mpk, msk) \leftarrow \mathtt{Setup}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$
  $(\sigma^*, \{L_i^*, R_i^* \mid i \in [1,D]\}, m^*) \leftarrow \mathcal{A}^{\mathfrak{Reveal},\mathfrak{Sign}}(mpk)$, where
  $-\mathfrak{Reveal}(\{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}, d_\iota)$: // $\iota \in [1, q_r]$
    Rtrn $sk_\iota \leftarrow \mathtt{KGen}(msk, \{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}, d_\iota)$.
  $-\mathfrak{Sign}(\{l_{i,\theta}, r_{i,\theta} \mid i \in [1,D]\}, d_\theta, \{L_{i,\theta}, R_{i,\theta} \mid i \in [1,D]\}, m_\theta \in \{0,1\}^*)$: // $\theta \in [1, q_s]$
    $sk_\theta \leftarrow \mathtt{KGen}(msk, \{l_{i,\theta}, r_{i,\theta} \mid i \in [1,D]\}, d_\theta)$.
    Rtrn $\sigma_\theta \leftarrow \mathtt{Sig}(sk_\theta, \{L_{i,\theta}, R_{i,\theta} \mid i \in [1,D]\}, m_\theta)$.
  Rtrn 1 if $1 \leftarrow \mathtt{Ver}(\sigma^*, \{L_i^*, R_i^* \mid i \in [1,D]\}, m^*)$
    $\bigwedge_{\iota \in [1,q_r]} \left(\sum_{i \in [1,D] \text{ s.t. } l_i^* \leq l_{i,\iota} \leq r_{i,\iota} \leq R_i^*} 1\right) < d_\iota$
    $\bigwedge_{\theta \in [1,q_s]} (\{L_{i,\theta}, R_{i,\theta} \mid i \in [1,D]\}, m_\theta) \neq (\{L_i^*, R_i^* \mid i \in [1,D]\}, m^*)$.
  Rtrn 0.

$\boldsymbol{Expt}^{\mathtt{PP}}_{\Sigma_{\mathrm{SP}},\mathcal{A},\beta}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$:   // $\beta \in \{0, 1\}$
  $(mpk, msk) \leftarrow \mathtt{Setup}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$.
  $(mpk, \widehat{msk}) \leftarrow \widehat{\mathtt{Setup}}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$.
  Rtrn $b \leftarrow \mathcal{A}^{\mathfrak{Reveal},\mathfrak{Sign}}(mpk, msk)$, where
  $-\mathfrak{Reveal}(\{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}, d_\iota)$: // $\iota \in [1, q_r]$
    Rtrn $\perp$ if $\neg \left[\bigwedge_{i \in [1,D]} 0 \leq l_{i,\iota} \leq r_{i,\iota} \leq T_i - 1\right] \bigvee d_\iota \notin [1,D]$.
    $sk_\iota \leftarrow \mathtt{KGen}(msk, \{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}, d_\iota)$.
    $sk_\iota \leftarrow \widehat{\mathtt{KGen}}(\widehat{msk}, \{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}, d_\iota)$. Rtrn $sk_\iota$.
  $-\mathfrak{Delegate}(\iota \in [1, q_r], \{l_i', r_i' \mid i \in [1,D]\})$:
    Rtrn $\perp$ if $\neg \left[\bigwedge_{i \in [1,D]} 0 \leq l_i' \leq l_{i,\iota} \leq r_{i,\iota} \leq r_i' \leq T_i - 1\right]$.
    For every $i \in [1,D]$, $(l_{i,\iota}, r_{i,\iota}) := (l_i', r_i')$.
    $sk_\iota' \leftarrow \mathtt{KDel}(sk_\iota, \{l_i', r_i' \mid i \in [1,D]\})$.
    $sk_\iota' \leftarrow \widehat{\mathtt{KDel}}(sk_\iota, \{l_i', r_i' \mid i \in [1,D]\})$. Rtrn $sk_\iota := sk_\iota'$.
  $-\mathfrak{Sign}(\iota \in [1, q_r], m, \{L_i, R_i \mid i \in [1,D]\})$:
    Rtrn $\perp$ if $\neg \left[\bigwedge_{i \in [1,D]} 0 \leq L_i \leq R_i \leq T_i - 1\right] \bigvee \left(\sum_{i \in [1,D] \text{ s.t. } L_i \leq l_{i,\iota} \leq r_{i,\iota} \leq R_i} 1\right) < d_\iota$.
    $\sigma \leftarrow \mathtt{Sig}(sk_\iota, \{L_i, R_i \mid i \in [1,D]\}, m)$.
    $\sigma \leftarrow \widehat{\mathtt{Sig}}(\widehat{msk}, \{L_i, R_i \mid i \in [1,D]\}, m)$. Rtrn $\sigma$.
```

Fig. 4. Security experiments w.r.t. an MDSBRS scheme Σ_{SB}. Top: (Adaptive) existential unforgeability. Bottom: Perfect privacy.

3.1 Informal Description of Our MDSBRS Scheme

IK-TSS Scheme. Our MDSBRS scheme is an extension of a TSS scheme proposed by Ishizaka and Kiyomoto [5] (denoted by IK-TSS). Let us firstly explain IK-TSS. A formal description of IK-TSS is presented in Sect. A.

IK-TSS is obtained by applying a generic approach (which transforms an FSS and BSS scheme into a TSS scheme) to a concrete FSS and BSS scheme (denoted by CS-FSS and CS-BSS, respectively) based on an HIBS scheme proposed by Chutterjee and Sarker [3] (denoted by CS-HIBS).

Let $\mathbb{S}(t)$ for $t \in [0, \log T-1]$ denote a set of identities $\{t\} \bigcup_{k \in [0, \log T_i - 1] \text{ s.t. } t[k]=0} \{t[0]||\cdots||t[k-1]||1\}$. Informally speaking, in CS-FSS, a secret-key for a time period $t \in [0, T-1]$ consists of randomly-generated CS-HIBS secret-keys for all identities in $\mathbb{S}(t)$. Specifically, it consists of $(g_1^\alpha \prod_{i \in [0, \log T-1]} (u_i v_0^{t[i]})^{r_i}, g^{r_0}, \cdots, g^{r_{\log T-1}}, \{g_1^\alpha \prod_{i \in [0,j-1]} (u_i v_0^{t[i]})^{r_i} (u_j v_0)^{r'_j}, g^{r'_j} \mid j \in [0, \log T-1] \text{ s.t. } t[j]=0\})$, where $g_1^\alpha \in \mathbb{G}$ is the master secret-key. If we have a secret-key for t, then we can generate one for any $t' > t$. For any identity $id \in \mathbb{S}(t')$, one of its ancestor identities must exist in $\mathbb{S}(t)$, which means that we can generate a CS-HIBS secret-key for id' from the one for the ancestor identity id. A signature on a message m under a time period t' consists of $(g_1^\alpha \prod_{i \in [0, \log T-1]} (u_i v_0^{t'[i]})^{r_i} (u \prod_{i \in [0, N-1]} v_i^{m[i]})^r$, $g^{r_0}, \cdots, g^{r_{\log T-1}}, g^r)$. If we have a correct secret-key sk_t for t, then we can generate a signature for any $t' > t$. We firstly transform sk_t into one for $sk_{t'}$ for t'. Generating a signature by using $sk_{t'}$ must be almost obvious.

Any FSS scheme can be easily transformed into a BSS scheme. In the BSS scheme, a secret-key (or signature) for t is identical to one for $\hat{t} := T-1-t \in [0, T-1]$ w.r.t. the underlying FSS scheme. CS-BSS is obtained by applying the transformation to CS-FSS. In CS-BSS, a secret-key for $t \in [0, T-1]$ (with $\hat{t} := T-1-t$) consists of $(g_1^\alpha \prod_{i \in [0, \log T-1]} (w_i v_0^{\hat{t}[i]})^{s_i}, g^{s_0}, \cdots, g^{s_{\log T-1}}$, $g_1^\alpha \prod_{i \in [0,j-1]} (w_i v_0^{\hat{t}[i]})^{s_i} (u_j v_0)^{s'_j}, g^{s'_j} \mid j \in [0, \log T-1] \text{ s.t. } \hat{t}[j]=0\})$. A signature on m under t' (with $\hat{t}' := T-1-t'$) consists of $(g_1^\alpha \prod_{i \in [0, \log T-1]} (w_i v_0^{\hat{t}'[i]})^{s_i} (u \prod_{i \in [0, N-1]} v_i^{m[i]})^s$, $g^{s_0}, \cdots, g^{s_{\log T-1}}, g^s)$.

As we mentioned earlier in Sect. 1, the *generic* method (where we independently generate an FSS and BSS secret-key for t and generate an FSS and BSS signature under R and L) is not secure against the colluding attacks. To make the method secure against such attasks, we divide the *true* master secret-key g_1^α into two shares of secret sharing, i.e., $g_1^\alpha g^\delta$ and $g^{-\delta}$ for $\delta \xleftarrow{\text{U}} \mathbb{Z}_p$. A signature on m under a range $[L, R]$ (with $\hat{L} := T-1-L$) consists of $(g_1^\alpha \prod_{i \in [0, \log T-1]} (u_i v_0^{R[i]})^{r_i} (w_i v_0^{\hat{L}[i]})^{s_i} (u \prod_{i \in [0, N-1]} v_i^{m[i]})^r, g^{r_0}, \cdots, g^{r_{\log T-1}}, g^{s_0}, \cdots$, $g^{s_{\log T-1}}, g^r)$. If and only if we have a correct FSS secret-key for t based on the first pseudo master secret-key $g_1^\alpha g^\delta$ and a correct BSS one for t based on the second pseudo master secret-key $g^{-\delta}$, we can generate a signature for $[L, R]$ s.t. $t \in [L, R]$.

Our MDSBRS Scheme. Our MDSBRS scheme is obtained by generalizing IK-TSS in the following 3 steps.

Firstly, generalizing the variable associated with a secret-key from a time period t to a range $[l, r]$ is straightforward. A secret-key for $[l, r]$ is composed of an FSS secret-key for r w.r.t. the CS-FSS scheme and a BSS one for l w.r.t. the CS-BSS scheme (or an FSS one for $\hat{l} := T - 1 - l$ w.r.t. the CS-FSS scheme).

Secondly, the current scheme has already implicitly been key-delegatable. Thus, a secret-key for $[l, r]$ can evolve into one for $[l', r']$ s.t. $l' \leq l \leq r \leq r'$ by transforming the FSS (resp. BSS) secret-key for r (resp. $\hat{l} := T - 1 - l$) into one for r' (resp. $\hat{l}' := T - 1 - l'$).

Thirdly, for the multi-dimensionalization with d-out-of-D threshold signability, we use the technique of d-out-of-D SSS. Firstly, we divide the true master-key g_1^α into D number of *first-level* pseudo master-keys $\{g_1^{f(i)} \mid i \in [1, D]\}$ based on the d-out-of-D SSS, where $f : [1, D] \rightarrow \mathbb{Z}_p$ is a randomly chosen $(d - 1)$-dimensional polynomial satisfying $f(0) = \alpha \in \mathbb{Z}_p$. Secondly, we divide each first-level pseudo master-key into 2 number of *second-level* pseudo master-keys $\{g_1^{f(i)} g^\delta, g^{-\delta} \mid i \in [1, D]\}$ based on the 2-out-of-2 SSS, where $\delta \overset{\mathrm{U}}{\leftarrow} \mathbb{Z}_p$. Specifically, a secret-key for $[l_i, r_i]$ consists of an FSS secret-key for r_i which consists of $(g_1^{f(i)} g^{\delta_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{r_i[j]})^{s_{ji}}, g^{s_{0,i}}, \cdots, g^{s_{\log T_i - 1, i}}, \{g_1^{f(i)} g^{\delta_i} \prod_{j \in [0, k-1]} (u_{ji} v_0^{r_i[j]})^{s_{ji}} (u_{ki} v_0)^{s'_{ki}}, g^{s'_{ki}} \mid k \in [0, \log T_i - 1] \text{ s.t. } r_i[k] = 0\})$ and an FSS one for $\hat{l}_i := T_i - 1 - l_i$ which consists of $(g^{-\delta_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{l}_i[j]})^{t_{ji}}, g^{t_{0,i}}, \cdots, g^{t_{\log T_i - 1, i}}, \{g^{-\delta_i} \prod_{j \in [0, k-1]} (w_{ji} v_0^{\hat{l}_i[j]})^{t_{ji}} (w_{ki} v_0)^{t'_{ki}}, g^{t'_{ki}} \mid k \in [0, \log T_i - 1] \text{ s.t. } \hat{l}_i[k] = 0\})$. A signature under $\{[L_i, R_i] \mid i \in [1, D]\}$ with $\hat{L}_i := T_i - 1 - L_i$ consists of

$$\Bigg(g_1^\alpha \prod_{i \in [1, D]} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s_{ji}} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji}} (u \prod_{j \in [0, N-1]} v_j^{m[j]})^r,$$
$$\{g^{s_{ji}}, g^{t_{ji}} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, g^r \Big) .$$

Let us consider a case where a user who has a correct secret-key associated with $\{[l_i, r_i] \mid i \in [1, D]\}$ and d would like to generate a signature under $\{[L_i, R_i] \mid i \in [1, D]\}$. If there eixsts a set of indices $\mathbb{I} = \{i \in [1, D] \text{ s.t. } L_i \leq l_i \leq r_i \leq R_i\}$ with $|\mathbb{I}| \geq d$, the user can correctly generate such a signature as follows. For $i \in \mathbb{I}$, we evolve the FSS secret-key for r_i into one for R_i, and evolve the FSS secret-key for $\hat{l}_i := T_i - 1 - l_i$ into one for $\hat{L}_i := T_i - 1 - L_i$. They are parsed as $(g_1^{f(i)} g^{\delta_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s_{ji}}, g^{s_{0,i}}, \cdots, g^{s_{\log T_i - 1, i}}, \cdots)$ and $(g^{-\delta_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji}}, g^{t_{0,i}}, \cdots, g^{t_{\log T_i - 1, i}}, \cdots)$, respectively. The SSS guarantees that we can derive $g_1^\alpha \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s_{ji}} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji}}$ by computing $\prod_{i \in \mathbb{I}} (D_{\log T_i, i} \cdot E_{\log T_i, i})^{\Delta_{i, \mathbb{I}}(0)}$, where $D_{\log T_i, i}$ (resp. $E_{\log T_i, i}$) is the first element of the secret-key for R_i (resp. L_i) and $\Delta_{i, \mathbb{I}}$ is the Lagrange coefficient.

3.2 Formal Description of Our MDSBRS Scheme

Our scheme $\Pi_{\mathrm{SB}} = \{\mathtt{Setup}, \mathtt{KGen}, \mathtt{KDel}, \mathtt{Sig}, \mathtt{Ver}\}$ is formally described as follows, where $\mathtt{KUpd}_\beta$ is an algorithm used as a sub-routine in $\mathtt{KDel}$.

$\texttt{Setup}(1^\lambda, D, T_1, \cdots, T_D)$**:** Let

- $(p, \mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T, e, g, \tilde{g}) \leftarrow \mathcal{G}_{BG}(1^\lambda)$. $\alpha \xleftarrow{\text{U}} \mathbb{Z}_p$, $g_2 := \tilde{g}^\alpha$. $g_1 \xleftarrow{\text{U}} \mathbb{G}$.
- For $i \in [1, D]$ and $j \in [0, \log T_i - 1]$: $x_{ji}, z_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$, $u_{ji} := g^{x_{ji}}, \tilde{u}_{ji} := \tilde{g}^{x_{ji}}, w_{ji} := g^{z_{ji}}, \tilde{w}_{ji} := \tilde{g}^{z_{ji}}$.
- $x \xleftarrow{\text{U}} \mathbb{Z}_p$, $u := g^x$, $\tilde{u} := \tilde{g}^x$.
- For $i \in [0, N-1]$: $y_i \xleftarrow{\text{U}} \mathbb{Z}_p$, $v_i := g^{y_i}, \tilde{v}_i := \tilde{g}^{y_i}$.
- $mpk := \left(\begin{array}{c} p, \mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T, e, g, \tilde{g}, g_1, g_2, \\ \left\{ u_{ji}, \tilde{u}_{ji}, w_{ji}, \tilde{w}_{ji} \middle| {\scriptstyle i \in [1,D], \atop j \in [0, \log T_i - 1]} \right\}, u, \tilde{u}, \{v_i, \tilde{v}_i \mid i \in [0, N-1]\} \end{array} \right)$.
- $msk := g_1^\alpha$.

It returns (mpk, msk).

$\texttt{KGen}(msk, l_1, r_1, \cdots, l_D, r_D, d)$**:** It returns $\perp$ if $\neg[1 \le d \le D \wedge_{i=1}^{D} 0 \le l_i \le r_i \le T_i - 1]$.

It chooses $A_1, \cdots, A_{d-1}$ uniformly at random from $\mathbb{Z}_p$. A $(d-1)$-dimensional polynomial $f : [1, D] \to \mathbb{Z}_p$ is defined as $f(x) := \sum_{j \in [1, d-1]} A_j x^j + \alpha$. For every $i \in [1, D]$, it does:

- $\delta_i \xleftarrow{\text{U}} \mathbb{Z}_p$. $\hat{l}_i := T_i - 1 - l_i$.
- For $j \in [0, \log T_i - 1]$: $s_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$. If $r_i[j] = 0$, $s'_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$.
- $sk_{r_i} := \left(\begin{array}{c} g_1^{f(i)} g^{\delta_i} \displaystyle\prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{r_i[j]})^{s_{ji}}, g^{s_{0,i}}, \cdots, g^{s_{\log T_i - 1, i}}, \\ \left\{ g_1^{f(i)} g^{\delta_i} \displaystyle\prod_{j \in [0, k-1]} (u_{ji} v_0^{r_i[j]})^{s_{ji}} (u_{ki} v_0)^{s'_{ki}}, g^{s'_{ki}} \middle| {\scriptstyle k \in [0, \log T_i - 1] \atop \text{s.t. } r_i[k] = 0} \right\} \end{array} \right)$.
- For $j \in [0, \log T_i - 1]$: $t_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$. If $\hat{l}_i[j] = 0$, $t'_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$.
- $sk_{l_i} := \left(\begin{array}{c} g^{-\delta_i} \displaystyle\prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{l}_i[j]})^{t_{ji}}, g^{t_{0,i}}, \cdots, g^{t_{\log T_i - 1, i}}, \\ \left\{ g^{-\delta_i} \displaystyle\prod_{j \in [0, k-1]} (w_{ji} v_0^{\hat{l}_i[j]})^{t_{ji}} (w_{ki} v_0)^{t'_{ki}}, g^{t'_{ki}} \middle| {\scriptstyle k \in [0, \log T_i - 1] \atop \text{s.t. } \hat{l}_i[k] = 0} \right\} \end{array} \right)$.

It returns $sk := (\{sk_{l_i}, sk_{r_i} \mid i \in [1, D]\}, d)$.

$\texttt{KUpd}_\beta(sk_r, r, R, i)$**:** $//\ \beta \in \{0, 1\}$

The algorithm updates a partial secret-key. $\texttt{KUpd}_0$ updates a partial right secret-key sk_r for $r \in [0, \log T_i - 1]$ in the i-th dimension to one for $R > r$. $\texttt{KUpd}_1$ updates a partial left secret-key.

It returns $\perp$ if $i \notin [1, D] \bigvee \neg [0 \le r \le R \le T_i - 1]$.

It parses sk_r as $(D_{\log T_i}, d_0, \cdots, d_{\log T_i - 1}, \{D_k, d'_k \mid k \in [0, \log T_i - 1] \text{ s.t. } r[k] = 0\})$. If $r < R$, then $\exists k_i \in [0, \log T_i - 1]$ s.t. $[k_i \neq 0 \implies \bigwedge_{j \in [0, k_i - 1]} r_i[j] = R_i[j]] \bigwedge [r_i[k_i] = 0 \bigwedge R_i[k_i] = 1]$. For every $j \in [k_i + 1, \log T_i - 1]$, $\tilde{s}_{k_i+1}, \cdots, \tilde{s}_{\log T_i - 1} \xleftarrow{\text{U}} \mathbb{Z}_p$. For every $k \in [k_i + 1, \log T_i - 1]$ s.t. $R[k] = 0$, $\tilde{s}'_k \xleftarrow{\text{U}} \mathbb{Z}_p$. It returns $sk_R := (\tilde{D}_{\log T_i}, \tilde{d}_0, \cdots, \tilde{d}_{\log T_i - 1}, \{\tilde{D}_k, \tilde{d}'_k \mid k \in [0, \log T_i - 1] \text{ s.t. } R[k] = 0\})$, where

- $\tilde{D}_{\log T_i} := \begin{cases} D_{k_i} \prod_{j \in [k_i+1, \log T_i - 1]} (u_{ji} v_0^{R[j]})^{\tilde{s}_j} & \text{(if } \beta = 0\text{)}, \\ D_{k_i} \prod_{j \in [k_i+1, \log T_i - 1]} (w_{ji} v_0^{R[j]})^{\tilde{s}_j} & \text{(otherwise)}. \end{cases}$
- For every $j \in [0, \log T_i - 1]$:

$$\tilde{d}_j := \begin{cases} d_j & \text{(if } j \in [0, k_i - 1]\text{)}, \\ d'_{k_i} & \text{(else if } j = k_i\text{)}, \\ g^{\tilde{s}_j} & \text{(otherwise)}. \end{cases}$$

- For every $k \in [0, \log T_i - 1]$ s.t. $R[k] = 0$:

$$(\tilde{D}_k, \tilde{d}'_k) := \begin{cases} (D_k, d'_k) & \text{(if } k < k_i\text{)}, \\ (D_{k_i} \prod_{j=k_i+1}^{k-1} (u_{ji} v_0^{R[j]})^{\tilde{s}_j} (u_{ki} v_0)^{\tilde{s}'_k}, g^{\tilde{s}'_k}) & \text{(else if } \beta = 0\text{)}, \\ (D_{k_i} \prod_{j=k_i+1}^{k-1} (w_{ji} v_0^{R[j]})^{\tilde{s}_j} (w_{ki} v_0)^{\tilde{s}'_k}, g^{\tilde{s}'_k}) & \text{(otherwise)}. \end{cases}$$

`KDel`$(sk, l'_1, r'_1, \cdots, l'_D, r'_D)$**:** It parses sk for $\{l_i, r_i \mid i \in [1, D]\}$ and d as $(\{sk_{l_i}, sk_{r_i} \mid i \in [1, D]\}, d)$. It returns $\perp$ if $\neg[\bigwedge_{i=1}^{D} 0 \le l'_i \le l_i \le r_i \le r'_i \le T_i - 1]$.
Firstly, it re-randomizes sk[4]. Let $A'_1, \cdots, A'_{d-1} \xleftarrow{\text{U}} \mathbb{Z}_p$. For every $i \in [1, D]$, it does:

- $\delta'_i \xleftarrow{\text{U}} \mathbb{Z}_p$. $\hat{l}_i := T_i - 1 - l_i$.
- Parse sk_{r_i} as $(D_{\log T_i, i}, d_{0,i}, \cdots, d_{\log T_i - 1, i}, \{D_{ki}, d'_{ki} \mid {}^{k \in [0, \log T_i - 1]}_{\text{s.t. } r_i[k]=0}\})$.
- For $j \in [0, \log T_i - 1]$: $\tilde{s}_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$. If $r_i[j] = 0$, $\tilde{s}'_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$.
- Let $\tilde{sk}_{r_i} := (\tilde{D}_{\log T_i, i}, \tilde{d}_{0,i}, \cdots, \tilde{d}_{\log T_i - 1, i}, \{\tilde{D}_{ki}, \tilde{d}'_{ki} \mid {}^{k \in [0, \log T_i - 1]}_{\text{s.t. } r_i[k]=0}\})$, where
  - $\tilde{D}_{\log T_i, i} := D_{\log T_i, i} \cdot g_1^{\sum_{j \in [1, d-1]} A'_j \cdot i^j} g^{\delta'_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{r_i[j]})^{\tilde{s}_{ji}}$.
  - $\tilde{d}_{ji} := d_{ji} \cdot g^{\tilde{s}_{ji}}$ (for each $j \in [0, \log T_i - 1]$).
  - $\tilde{D}_{ki} := D_{ki} \cdot g_1^{\sum_{j \in [1, d-1]} A'_j \cdot i^j} g^{\delta'_i} \prod_{j \in [0, k-1]} (u_{ji} v_0^{r_i[j]})^{\tilde{s}_{ji}} (u_{ki} v_0)^{\tilde{s}'_{ki}}$ and $\tilde{d}'_{ki} := d'_{ki} \cdot g^{\tilde{s}'_{ki}}$ (for each $k \in [0, \log T_i - 1]$ s.t. $r_i[k] = 0$).
- Parse sk_{l_i} as $(E_{\log T_i, i}, e_{0,i}, \cdots, e_{\log T_i - 1, i}, \{E_{ki}, e'_{ki} \mid {}^{k \in [0, \log T_i - 1]}_{\text{s.t. } \hat{l}_i[k]=0}\})$.
- For $j \in [0, \log T_i - 1]$: $\tilde{t}_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$. If $\hat{l}_i[j] = 0$, $\tilde{t}'_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$.
- Let $\tilde{sk}_{l_i} := (\tilde{E}_{\log T_i, i}, \tilde{e}_{0,i}, \cdots, \tilde{e}_{\log T_i - 1, i}, \{\tilde{E}_{k,i}, \tilde{e}'_{k,i} \mid {}^{k \in [0, \log T_i - 1]}_{\text{s.t. } \hat{l}_i[k]=0}\})$, where
  - $\tilde{E}_{\log T_i, i} := E_{\log T_i, i} \cdot g^{-\delta'_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{l}_i[j]})^{\tilde{t}_{ji}}$.
  - $\tilde{e}_{ji} := e_{ji} \cdot g^{\tilde{t}_{ji}}$ (for each $j \in [0, \log T_i - 1]$).
  - $\tilde{E}_{ki} := E_{ki} \cdot g^{-\delta'_i} \prod_{j \in [0, k-1]} (w_{ji} v_0^{\hat{l}_i[j]})^{\tilde{t}_{ji}} (w_{ki} v_0)^{\tilde{t}'_{ki}}$ and $\tilde{e}'_{ki} := e'_{ki} \cdot g^{\tilde{t}'_{ki}}$ (for each $k \in [0, \log T_i - 1]$ s.t. $\hat{l}_i[k] = 0$).

Secondly, it updates the re-randomized secret-key for $\{l_i, r_i \mid i \in [1, D]\}$ to one for $\{l'_i, r'_i \mid i \in [1, D]\}$. For each $i \in [1, D]$ s.t. $r_i < r'_i$, it updates the partial secret-key $\tilde{sk}_{r_i}$ for r_i to one for r'_i by $sk_{r'_i} \leftarrow \texttt{KUpd}_0(\tilde{sk}_{r_i}, r_i, r'_i, i)$. For each $i \in [1, D]$ s.t. $l_i > l'_i$, it updates $\tilde{sk}_{l_i}$ for l_i to one for l'_i by $sk_{l'_i} \leftarrow \texttt{KUpd}_1(\tilde{sk}_{l_i}, \hat{l}_i, \hat{l}'_i, i)$, where $\hat{l}'_i := T_i - 1 - l'_i$.
Finally, it returns $sk' := (\{sk_{l'_i}, sk_{r'_i} \mid i \in [1, D]\}, d)$.

[4] `KDel` is used as a sub-routine in `Sig`. The key re-randomization is necessary for the scheme to achieve perfect privacy.

$\texttt{Sig}(sk, L_1, R_1, \cdots, L_D, R_D, m)$**:** It parses sk for $\{l_i, r_i \mid i \in [1, D]\}$ and d as $(\{sk_{l_i}, sk_{r_i} \mid i \in [1, D]\}, d)$. It returns $\perp$ if $\neg \left[\bigwedge_{i \in [1,D]} 0 \leq L_i \leq R_i \leq T_i - 1 \wedge \sum_{i \in [1,D] \text{ s.t. } L_i \leq l_i \leq r_i \leq R_i} 1 \geq d\right]$.

Let $\mathbb{I}$ be a set $\{i \in [1, D] \text{ s.t. } L_i \leq l_i \leq r_i \leq R_i\}$ satisfying $|\mathbb{I}| \geq d$. For each $i \in \mathbb{I}$, let $(l'_i, r'_i) := (L_i, R_i)$. For each $i \in [1, D] \setminus \mathbb{I}$, let $(l'_i, r'_i) := (l_i, r_i)$. Firstly, from sk, it generates a delegated and re-randomized secret-key for $\{l'_i, r'_i \mid i \in [1, D]\}$. Thus, $\tilde{sk} \leftarrow \texttt{KDel}(sk, \{l'_i, r'_i \mid i \in [1, D]\})$. $\tilde{sk}$ is parsed as $(\{sk_{l'_i}, sk_{r'_i} \mid i \in [1, D]\})$[5].

For every $i \in \mathbb{I}$, it does:

- Parse $sk_{r'_i}$ as $(D_{\log T_i, i}, d_{0,i}, \cdots, d_{\log T_i - 1, i}, \cdots)$.
- Parse $sk_{l'_i}$ as $(E_{\log T_i, i}, e_{0,i}, \cdots, e_{\log T_i - 1, i}, \cdots)$.
- Define a function $\Delta_{i,\mathbb{I}}(x) := \prod_{j \in \mathbb{I} \setminus \{i\}} \frac{x-j}{i-j}$.

For every $i \in [1, D] \setminus \mathbb{I}$, it does:

- $\hat{L}_i := T_i - 1 - L_i$.
- For every $j \in [0, \log T_i - 1]$, $s^*_{j,i}, t^*_{j,i} \xleftarrow{\text{U}} \mathbb{Z}_p$.

It chooses $r \xleftarrow{\text{U}} \mathbb{Z}_p$. It returns $\sigma := (U, \{V_{ji}, V'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, W)$, where

- $U := \prod_{i \in \mathbb{I}} (D_{\log T_i, i} \cdot E_{\log T_i, i})^{\Delta_{i,\mathbb{I}}(0)}$
  $\cdot \prod_{i \in [1,D] \setminus \mathbb{I}} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s^*_{ji}} (w_{ji} v_0^{\hat{L}_i[j]})^{t^*_{ji}} \cdot (u \prod_{j \in [0, N-1]} v_j^{m[j]})^r$,
- $V_{ji} := (d_{ji})^{\Delta_{i,\mathbb{I}}(0)}$ and $V'_{ji} := (e_{ji})^{\Delta_{i,\mathbb{I}}(0)}$ (for $i \in \mathbb{I}$ and $j \in [0, \log T_i - 1]$),
- $V_{ji} := g^{s^*_{ji}}$ and $V'_{ji} := g^{t^*_{ji}}$ (for $i \in [1, D] \setminus \mathbb{I}$ and $j \in [0, \log T_i - 1]$),
- $W := g^r$.

$\texttt{Ver}(\sigma, L_1, R_1, \cdots, L_D, R_D, m)$**:** It parses σ as $(U, \{V_{ji}, V'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, W)$. It returns 1 if it holds that

$$e(U, \tilde{g}) = e(g_1, g_2) \cdot \prod_{i \in [1,D]} \prod_{j \in [0, \log T_i - 1]} e(V_{ji}, \tilde{u}_{ji} \tilde{v}_0^{R_i[j]}) e(V'_{ji}, \tilde{w}_{ji} \tilde{v}_0^{\hat{L}_i[j]})$$
$$\cdot e(W, \tilde{u} \prod_{j \in [0, N-1]} \tilde{v}_j^{m[j]}).$$

It returns 0 otherwise.

3.3 Existential Unforgeability of Our MDSBRS Scheme

Theorem 1. *Π_{SB} is existentially unforgeable under the co-CDH assumption.*

[5] Among the partial secret-keys, we only use $\{sk_{l'_i}, sk_{r'_i} \mid i \in \mathbb{I}\}$ to generate a signature. Thus, we do not need to generate $\{sk_{l'_i}, sk_{r'_i} \mid i \in [1, D] \setminus \mathbb{I}\}$. Moreover, for each partial secret-key in $\{sk_{l'_i}, sk_{r'_i} \mid i \in \mathbb{I}\}$, we use only the first $\log T_i + 1$ elements, which means the other elements are unnecessary.

Proof. Let $\mathcal{A} \in \mathbb{PPT}_\lambda$ denote a PPT algorithm which behaves as an adversary in the existential unforgeability experiment in Fig. 4 w.r.t. the scheme Π_{SB}. We prove that there exists another PPT algorithm $\mathcal{B} \in \mathbb{PPT}_\lambda$ which uses $\mathcal{A}$ as a sub-routine to break the co-CDH assumption with

$$\mathtt{Adv}_{\mathcal{B}}^{\mathtt{co\text{-}CDH}}(\lambda) \geq \frac{\mathtt{Adv}_{\Pi_{\mathrm{SB}},\mathcal{A},D,T_1,\cdots,T_D,N}^{\mathtt{EUF\text{-}CMA}}(\lambda)}{2\left\{2\left(\left(\sum_{i=1}^{D}\log T_i\right)q_r + q_s\right)(N+1)\right\}^{2\left(\sum_{i=1}^{D}\log T_i\right)+1}}. \qquad (1)$$

If we introduce the following two restrictions, (i) every threshold $d_\iota \in [1, D]$ queried to $\mathfrak{Reveal}$ is D and (ii) there exists an integer $T \in \mathbb{N}$, for every $i \in [1, D]$, T_i is equal to T, we obtain a conciser result given below.

$$\mathtt{Adv}_{\mathcal{B}}^{\mathtt{co\text{-}CDH}}(\lambda) \geq \frac{\mathtt{Adv}_{\Pi_{\mathrm{SB}},\mathcal{A},D,T_1,\cdots,T_D,N}^{\mathtt{EUF\text{-}CMA}}(\lambda)}{2\left\{2\left(\log T \cdot q_r + q_s\right)(N+1)\right\}^{2D\log T+1}}. \qquad (2)$$

We let $\mathcal{B}$ behave as follows.

$\mathcal{B}$ is given $(p, \mathbb{G}, \tilde{\mathbb{G}}, g, \tilde{g}, g^\beta, g^\alpha, \tilde{g}^\alpha)$ as an instance for the co-CDH assumption. $\mathcal{B}$ sets $g_1 := g^\beta$ and $g_2 := \tilde{g}^\alpha$. $\mathcal{B}$ chooses an integer n satisfying $n(N+1) < p$. $\mathcal{B}$ randomly chooses the following variables: $\{k_{ji}, s_{ji} \xleftarrow{\mathrm{U}} [0, N], x_{ji}, z_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_n, x'_{ji}, z'_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p \mid i \in [1, D], j \in [0, \log T_i - 1]\}$. $\hat{k} \xleftarrow{\mathrm{U}} [0, N], \hat{x} \xleftarrow{\mathrm{U}} \mathbb{Z}_n, \hat{x}' \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. $\left\{y_i \xleftarrow{\mathrm{U}} \mathbb{Z}_n, y'_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p \mid i \in [0, N-1]\right\}$.

$\mathcal{B}$ computes the following variables and sends mpk to $\mathcal{A}$.

- $\left\{u_{ji} := (g^\alpha)^{p-nk_{ji}+x_{ji}}\, g^{x'_{ji}}, \tilde{u}_{ji} := (\tilde{g}^\alpha)^{p-nk_{ji}+x_{ji}}\, \tilde{g}^{x'_{ji}} \mid i \in [1, D], j \in [0, \log T_i - 1]\right\}$.
- $\left\{w_{ji} := (g^\alpha)^{p-ns_{ji}+z_{ji}}\, g^{z'_{ji}}, \tilde{w}_{ji} := (\tilde{g}^\alpha)^{p-ns_{ji}+z_{ji}}\, \tilde{g}^{z'_{ji}} \mid i \in [1, D], j \in [0, \log T_i - 1]\right\}$.
- $u := (g^\alpha)^{p-n\hat{k}+\hat{x}}\, g^{\hat{x}'}, \tilde{u} := (\tilde{g}^\alpha)^{p-n\hat{k}+\hat{x}}\, \tilde{g}^{\hat{x}'}$.
- $\left\{v_i := (g^\alpha)^{y_i}\, g^{y'_i}, \tilde{v}_i := (\tilde{g}^\alpha)^{y_i}\, \tilde{g}^{y'_i} \mid i \in [0, N-1]\right\}$.
- $mpk := \left(\begin{array}{c} p, \mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T, e, g, \tilde{g}, g_1, g_2, \\ \left\{u_{ji}, \tilde{u}_{ji}, w_{ji}, \tilde{w}_{ji} \middle| {\scriptstyle i\in[1,D], \atop j\in[0,\log T_i-1]}\right\}, u, \tilde{u}, \{v_i, \tilde{v}_i \mid i \in [0, N-1]\} \end{array}\right)$

$\mathcal{B}$ defines the following functions.

- For $i \in [1, D]$, $j \in [0, \log T_i - 1]$ and a bit $b \in \{0, 1\}$,

$$\mathbf{F}_{ji}(b) := p - nk_{ji} + x_{ji} + y_0 b, \mathbf{J}_{ji}(b) := x'_{ji} + y'_0 b,$$

$$\mathbf{L}_{ji}(b) := x_{ji} + y_0 b \bmod n, \text{ and } \mathbf{K}_{ji}(b) := \begin{cases} 0 & \text{if } \mathbf{L}_{ji}(b) = 0, \\ 1 & \textit{otherwise}. \end{cases}$$

- For $i \in [1, D]$, $j \in [0, \log T_i - 1]$ and a bit $b \in \{0, 1\}$,

$$\mathbf{H}_{ji}(b) := p - ns_{ji} + z_{ji} + y_0 b, \mathbf{Q}_{ji}(b) := z'_{ji} + y'_0 b,$$

$$\mathbf{R}_{ji}(b) := z_{ji} + y_0 b \bmod n, \text{ and } \mathbf{U}_{ji}(b) := \begin{cases} 0 & \text{if } \mathbf{R}_{ji}(b) = 0, \\ 1 & \textit{otherwise}. \end{cases}$$

- For $m \in \{0,1\}^N$,

$$\mathbf{F}(m) := p - n\hat{k} + \hat{x} + \sum_{i\in[0,N-1]} y_i m[i], \mathbf{J}(m) := \hat{x}' + \sum_{i\in[0,N-1]} y'_i m[i]$$

$$\mathbf{L}(m) := \hat{x} + \sum_{i\in[0,N-1]} y_i m[i] \bmod n, \text{ and } \mathbf{K}(m) := \begin{cases} 0 & \text{if } \mathbf{L}(m) = 0, \\ 1 & \textit{otherwise.} \end{cases}$$

Let us consider a case where $\mathcal{A}$ issues $\{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}$ as the ι-th query to $\mathfrak{Reveal}$. Let $\hat{l}_{i,\iota} := T_i - 1 - l_{i,\iota}$. $\mathcal{B}$ defines a set $\mathbb{I}_\iota \subseteq [1,D]$ as

$$\begin{aligned} &\mathbb{I}_\iota := \{ i \in [1,D] \text{ s.t.} \\ &\bigvee_{\substack{j\in[0,\log T_i-1] \\ \text{s.t. } r_{i,\iota}[j]=1}} \left[\mathbf{K}_{ji}(1) = 1 \bigwedge \left[j \neq 0 \implies \bigwedge_{\substack{k\in[0,j-1] \\ \text{s.t. } r_{i,\iota}[k]=0}} \mathbf{K}_{ki}(1) = 1 \right]\right] \\ &\bigvee_{\substack{j\in[0,\log T_i-1] \\ \text{s.t. } \hat{l}_{i,\iota}[j]=1}} \left[\mathbf{U}_{ji}(1) = 1 \bigwedge \left[j \neq 0 \implies \bigwedge_{\substack{k\in[0,j-1] \\ \text{s.t. } \hat{l}_{i,\iota}[k]=0}} \mathbf{U}_{ki}(1) = 1 \right]\right] \} . \end{aligned} \tag{3}$$

$\mathcal{B}$ takes different actions in the following two cases:

$$\text{(R1) } |\mathbb{I}_\iota| \geq D - d_\iota + 1, \text{ and (R2) } \textit{Otherwise.}$$

Specifically, $\mathcal{B}$ behaves as follows in each case.

The case R1: If $|\mathbb{I}_\iota| = D - d_\iota + 1$, let $\mathbb{I}^* := \mathbb{I}_\iota$. Else, $\mathcal{B}$ arbitrarily chooses a subset $\mathbb{I}^*$ with cardinality $D - d_\iota + 1$ from $\mathbb{I}_\iota$.

For every $i \in [1,D] \setminus \mathbb{I}^*$, $\mathcal{B}$ randomly chooses $B_i \xleftarrow{\text{U}} \mathbb{Z}_p$. $\mathcal{B}$ assumes that

$$(f(i) =) \sum_{j\in[1,d_\iota-1]} i^j \cdot A_j + \alpha =: B_i \tag{4}$$

for unknown variables $A_1, \cdots, A_{d_\iota-1}, \alpha \in \mathbb{Z}_p$. For every $i \in \mathbb{I}^*$, $\mathcal{B}$ considers some variables C_i and $\{C_{ji} \mid j \in [1,D] \setminus \mathbb{I}^*\}$, and assumes that

$$(f(i) =) \sum_{j\in[1,d_\iota-1]} i^j \cdot A_j + \alpha =: \sum_{j\in[1,D]\setminus\mathbb{I}^*} C_{ji} B_j + C_i \alpha. \tag{5}$$

The variables $\{C_i \mid i \in \mathbb{I}^*\}$ and $\{C_{ji} \mid i \in \mathbb{I}^*, j \in [1,D] \setminus \mathbb{I}^*\}$ which satisfy both of the equations (4) and (5) uniquely exist, and they can be efficiently derived. Here, $\mathcal{B}$ derives them.

We remind us that a secret-key sk_ι for key-ranges $\{l_{i,\iota}, r_{i,\iota} \mid i \in [1,D]\}$ and a threshold d_ι consists of $2D$ partial secret-keys $\{sk'_{i,\iota}, sk_{i,\iota} \mid i \in [1,D]\}$, where $sk_{i,\iota}$ (resp. $sk'_{i,\iota}$) is the one for $r_{i,\iota}$ (resp. $l_{i,\iota}$).

For $i \in [1,D] \setminus \mathbb{I}^*$, $\mathcal{B}$ generates the partial secret-keys $(sk'_{i,\iota}, sk_{i,\iota})$ as follows.

- $\delta_i \xleftarrow{\text{U}} \mathbb{Z}_p$.
- For $j \in [0, \log T_i - 1]$: $s_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$. If $r_{i,\iota}[j] = 0$, $s'_{ji} \xleftarrow{\text{U}} \mathbb{Z}_p$.

- $sk_{i,\iota} := \left(\begin{array}{c} g_1^{B_i} g^{\delta_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{r_{i,\iota}[j]})^{s_{ji}}, g^{s_{0,i}}, \cdots, g^{s_{\log T_i - 1, i}}, \\ \left\{ g_1^{B_i} g^{\delta_i} \prod_{j \in [0, k-1]} (u_{ji} v_0^{r_{i,\iota}[j]})^{s_{ji}} (u_{ki} v_0)^{s'_{ki}}, g^{s'_{ki}} \middle| \begin{array}{c} k \in [0, \log T_i - 1] \\ \text{s.t. } r_{i,\iota}[k] = 0 \end{array} \right\} \end{array} \right).$
- For $j \in [0, \log T_i - 1]$: $t_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. If $\hat{l}_{i,\iota}[j] = 0$, $t'_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $sk'_{i,\iota} := \left(\begin{array}{c} g^{-\delta_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{l}_{i,\iota}[j]})^{t_{ji}}, g^{t_{0,i}}, \cdots, g^{t_{\log T_i - 1, i}}, \\ \left\{ g^{-\delta_i} \prod_{j \in [0, k-1]} (w_{ji} v_0^{\hat{l}_{i,\iota}[j]})^{t_{ji}} (w_{ki} v_0)^{t'_{ki}}, g^{t'_{ki}} \middle| \begin{array}{c} k \in [0, \log T_i - 1] \\ \text{s.t. } \hat{l}_{i,\iota}[k] = 0 \end{array} \right\} \end{array} \right).$

For $i \in \mathbb{I}^*$, $\mathcal{B}$ generates the partial secret-keys $(sk'_{i,\iota}, sk_{i,\iota})$ in a slightly-complicated manner. We consider the following three subcases:

$$\text{(R11)} \quad \bigvee_{\substack{j \in [0, \log T_i - 1] \\ \text{s.t. } r_{i,\iota}[j] = 1}} \left[\mathbf{K}_{ji}(1) = 1 \bigwedge \left[j \neq 0 \implies \bigwedge_{\substack{k \in [0, j-1] \\ \text{s.t. } r_{i,\iota}[k] = 0}} \mathbf{K}_{ki}(1) = 1 \right] \right],$$

$$\text{(R12)} \quad \bigvee_{\substack{j \in [0, \log T_i - 1] \\ \text{s.t. } \hat{l}_{i,\iota}[j] = 1}} \left[\mathbf{U}_{ji}(1) = 1 \bigwedge \left[j \neq 0 \implies \bigwedge_{\substack{k \in [0, j-1] \\ \text{s.t. } \hat{l}_{i,\iota}[k] = 0}} \mathbf{U}_{ki}(1) = 1 \right] \right],$$

(R13) *Otherwise.*

In each case, $\mathcal{B}$ behaves as follows.

The subcase R11: $\mathcal{B}$ generates the partial secret-keys $(sk'_{i,\iota}, sk_{i,\iota})$ as follows.

$\mathcal{B}$ firstly generates $sk_{i,\iota} = (D_{\log T_i, i}, d_{0,i}, \cdots, d_{\log T_i - 1, i}, \{D_{ki}, d'_{ki} \mid k \in [0, \log T_i - 1]$ s.t. $r_{i,\iota}[k] = 0\})$ as follows.

Let $\hat{j} \in [0, \log T_i - 1]$ denote an integer j which satisfies the condition for the subcase R11. It is implied that

$$r_{i,\iota}[\hat{j}] = 1 \bigwedge \left[\mathbf{F}_{\hat{j},i}(1) \neq 0 \bigwedge \left[\hat{j} \neq 0 \implies \bigwedge_{\substack{k \in [0, \hat{j}-1] \\ \text{s.t. } r_{i,\iota}[k] = 0}} \mathbf{F}_{k,i}(1) \neq 0 \right] \right].$$

Let $\delta_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. For every $j \in [0, \log T_i - 1]$, let $s_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. For every $j \in [0, \log T_i - 1]$, $\mathcal{B}$ computes

$$d_{ji} := \begin{cases} g_1^{-1/\mathbf{F}_{\hat{j},i}(1)} g^{s_{\hat{j},i}} & (\text{if } j = \hat{j}), \\ g^{s_{ji}} & (\text{otherwise}), \end{cases}$$

$$\varDelta_{ji} := \begin{cases} g_1^{-\mathbf{J}_{\hat{j},i}(1)/\mathbf{F}_{\hat{j},i}(1)} (g^{\alpha})^{s_{\hat{j},i}\mathbf{F}_{\hat{j},i}(1)} g^{s_{\hat{j},i}\mathbf{J}_{\hat{j},i}(1)} & (\text{if } j = \hat{j}), \\ (u_{ji} v_0^{r_{i,\iota}[j]})^{s_{ji}} & (\text{otherwise}). \end{cases}$$

Using them, $\mathcal{B}$ computes

$$D_{\log T_i,i} := g^{\delta_i} \cdot \prod_{j\in[0,\log T_i-1]} \varDelta_{ji}.$$

Note that $d_{\hat{j},i}$ and $\varDelta_{\hat{j},i}$ distribute identically to g^s and $g_1^{\alpha}(u_{\hat{j},i}v_0)^s$ for $s \xleftarrow{\mathrm{U}} \mathbb{Z}_p$ respectively since

$$\begin{aligned} d_{\hat{j},i} &= g^{s_{\hat{j},i}-\beta/\mathbf{F}_{\hat{j},i}(1)} =: g^{\tilde{s}_{\hat{j},i}} \text{ (where } \tilde{s}_{\hat{j},i} := s_{\hat{j},i} - \beta/\mathbf{F}_{\hat{j},i}(1)), \\ \varDelta_{\hat{j},i} &= g_1^{\alpha} g_1^{-\alpha\mathbf{F}_{\hat{j},i}(1)/\mathbf{F}_{\hat{j},i}(1)} g_1^{-\mathbf{J}_{\hat{j},i}(1)/\mathbf{F}_{\hat{j},i}(1)} g^{s_{\hat{j},i}(\alpha\mathbf{F}_{\hat{j},i}(1)+\mathbf{J}_{\hat{j},i}(1))} \\ &= g_1^{\alpha} g^{-\frac{\beta}{\mathbf{F}_{\hat{j},i}(1)}(\alpha\mathbf{F}_{\hat{j},i}(1)+\mathbf{J}_{\hat{j},i}(1))} g^{s_{\hat{j},i}(\alpha\mathbf{F}_{\hat{j},i}(1)+\mathbf{J}_{\hat{j},i}(1))} \\ &= g_1^{\alpha} g^{(s_{\hat{j},i}-\frac{\beta}{\mathbf{F}_{\hat{j},i}(1)})(\alpha\mathbf{F}_{\hat{j},i}(1)+\mathbf{J}_{\hat{j},i}(1))} = g_1^{\alpha} g^{\tilde{s}_{\hat{j},i}(\alpha\mathbf{F}_{\hat{j},i}(1)+\mathbf{J}_{\hat{j},i}(1))} \\ &= g_1^{\alpha} g^{\tilde{s}_{\hat{j},i}(\alpha(p-nk_{\hat{j},i}+x_{\hat{j},i}+y_0)+x'_{\hat{j},i}+y'_0)} \\ &= g_1^{\alpha}\left((g^{\alpha})^{p-nk_{\hat{j},i}+x_{\hat{j},i}} g^{x'_{\hat{j},i}} (g^{\alpha})^{y_0} g^{y'_0}\right)^{\tilde{s}_{\hat{j},i}} = g_1^{\alpha}\left(u_{\hat{j},i}v_0\right)^{\tilde{s}_{\hat{j},i}}. \end{aligned}$$

For every $k \in [\hat{j}+1, \log T_i - 1]$ s.t. $r_{i,\iota}[k] = 0$, $\mathcal{B}$ chooses $s'_{ki} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$ and computes

$$\begin{aligned} d'_{ki} &:= g^{s'_{ki}}, \\ D_{ki} &:= g^{\delta} \prod_{j\in[0,\hat{j}]} \varDelta_{ji} \prod_{j\in[\hat{j}+1,k-1]} (u_{ji} v_0^{r_{i,\iota}[j]})^{s_{ji}} (u_{ki}v_0)^{s'_{ki}}. \end{aligned}$$

If a statement $\hat{j} \neq 0 \bigwedge \exists k \in [0, \hat{j} - 1]$ s.t. $r_{i,\iota}[k] = 0$ is logically true, then for every $k \in [0, \hat{j} - 1]$ s.t. $r_{i,\iota}[k] = 0$, $\mathcal{B}$ computes

$$\begin{aligned} d'_{ki} &:= g_1^{-1/\mathbf{F}_{k,i}(1)} g^{s'_{ki}}, \\ \varDelta'_{ki} &:= g_1^{-\mathbf{J}_{k,i}(1)/\mathbf{F}_{k,i}(1)} (g^{\alpha})^{s'_{ki}\mathbf{F}_{k,i}(1)} g^{s'_{k,i}\mathbf{J}_{k,i}(1)} g^{\delta_i} \end{aligned}$$

$\mathcal{B}$ computes

$$D_{ki} := g^{\delta_i} \varDelta'_{ki} \prod_{j\in[0,k-1]} \varDelta_{ji}.$$

The fact that d'_{ki} and $\varDelta'_{ki}$ distribute identically to g^s and $g_1^{\alpha}(u_{\hat{j},i}v_0)^s$ for $s \xleftarrow{\mathrm{U}} \mathbb{Z}_p$ respectively can be verified in the same manner as the above $d_{\hat{j},i}$ and $\varDelta_{\hat{j},i}$. $\mathcal{B}$ finally updates the partial secret-key $sk_{i,\iota}$ to

$$\left(\begin{array}{c} D_{\log T_i,i}^{C_i} \cdot g_1^{\sum_{j\in[1,D]\setminus\mathbb{I}^*} C_{ji}B_j}, d_{0,i}^{C_i}, \cdots, d_{\log T_i-1,i}^{C_i}, \\ \left\{ (D_{ki})^{C_i} \cdot g_1^{\sum_{j\in[1,D]\setminus\mathbb{I}^*} C_{ji}B_j}, (d'_{ki})^{C_i} \mid k \in [0, \log T_i - 1] \text{ s.t. } r_{i,\iota}[k] = 0 \right\} \end{array} \right).$$

Next, $\mathcal{B}$ generates $sk'_{i,\iota} = (E_{\log T_i,i}, e_{0,i}, \cdots, e_{\log T_i - 1,i}, \{E_{ki}, e'_{ki} \mid k \in [0, \log T_i - 1] \text{ s.t. } \hat{l}_{i,\iota}[k] = 0\})$ as follows.

For every $j \in [0, \log T_i - 1]$, $t_{ji} \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. For every $k \in [0, \log T_i - 1]$ s.t. $\hat{l}_{i,\iota}[k] = 0$, $t'_{ki} \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. $\mathcal{B}$ computes

$$\begin{aligned} E_{\log T_i,i} &:= g^{-\delta_i} \prod_{j\in[0,\log T_i - 1]} (w_{ji} v_0^{\hat{l}_{i,\iota}[j]})^{t_{ji}}, \\ e_{ji} &:= g^{t_{ji}} \quad (\text{for } j \in [0, \log T_i - 1]), \\ (E_{ki}, e'_{ki}) &:= \left(g^{-\delta_i} \prod_{j\in[0,k-1]} (w_{ji} v_0^{\hat{l}_{i,\iota}[j]})^{t_{ji}} (w_{ki} v_0)^{t'_{ki}}, g^{t'_{ki}} \right) \\ &\qquad (\text{for } k \in [0, \log T_i - 1] \text{ s.t. } \hat{l}_{i,\iota}[k] = 0). \end{aligned}$$

$\mathcal{B}$ finally updates the partial secret-key $sk'_{i,\iota}$ to

$$\left(E^{C_i}_{\log T_i,i}, e^{C_i}_{0,i}, \cdots, e^{C_i}_{\log T_i - 1,i}, \{E^{C_i}_{ki}, (e'_{ki})^{C_i} \mid k \in [0, \log T_i - 1] \text{ s.t. } \hat{l}_{i,\iota}[k] = 0\} \right).$$

The subcase R12: $\mathcal{B}$ behaves analogously to the case R11. Because of the redundancy, we omit to describe it.

The subcase R13: $\mathcal{B}$ aborts the simulation.

If, for every $i \in [1, D]$, $\mathcal{B}$ successfully generates $(sk'_{i,\iota}, sk_{i,\iota})$, then $\mathcal{B}$ returns the secret-key $sk_\iota := (\{sk'_{i,\iota}, sk_{i,\iota} \mid i \in [1, D]\}, d_\iota)$ to $\mathcal{A}$. The secret-key correctly distributes.

The case R2: $\mathcal{B}$ aborts the simulation.

When $\mathcal{A}$ issues $(\{l_{i,\theta}, r_{i,\theta} \mid i \in [1, D]\}, \{L_{i,\theta}, R_{i,\theta} \mid i \in [1, D]\}, m_\theta)$ as the θ-th query to $\mathfrak{Sign}$, $\mathcal{B}$ takes different actions in the following four cases:

$$\begin{aligned} &(\text{S1}) \bigvee_{i\in[1,D]} \bigvee_{j\in[0,\log T_i - 1]} \mathbf{K}_{ji}(R_{i,\theta}[j]) = 1, \\ &(\text{S2}) \bigvee_{i\in[1,D]} \bigvee_{j\in[0,\log T_i - 1]} \mathbf{U}_{ji}(\hat{L}_{i,\theta}[j]) = 1, \\ &(\text{S3})\ \mathbf{K}(m_\theta) = 1, \text{ and (S4) } \textit{Otherwise}, \end{aligned}$$

where $\hat{L}_{i,\theta} := T_i - 1 - L_{i,\theta}$.

The case S1: According to the definition of the case, there exists $\hat{\imath} \in [1, D]$ and $\hat{\jmath} \in [0, \log T_{\hat{\imath}} - 1]$ such that $\mathbf{K}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\jmath}]) = 1$, which implies that $\mathbf{F}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\jmath}]) \neq 0$.

Let $r \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. For every $i \in [1, D]$ and $j \in [0, \log T_i - 1]$, let $s_{ji}, t_{ji} \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. $\mathcal{B}$ computes

$$U := \mathit{\Delta}_{\hat{\jmath},\hat{\imath}} \prod_{j\in[0,\log T_{\hat{\imath}} - 1]\setminus\{\hat{\jmath}\}} \left(u_{j,\hat{\imath}} v_0^{R_{\hat{\imath},\theta}[j]} \right)^{s_{j,\hat{\imath}}} \cdot \prod_{i\in[1,D]\setminus\{\hat{\imath}\}} \prod_{j\in[0,\log T_i - 1]} \left(u_{ji} v_0^{R_{i,\theta}[j]} \right)^{s_{ji}}$$

$$
\begin{aligned}
&\qquad \prod_{i\in[1,D]}\prod_{j\in[0,\log T_i-1]}\left(w_{ji}v_0^{\hat{L}_{i,\theta}[j]}\right)^{t_{ji}}\left(u\prod_{i\in[0,N-1]}v_i^{m_\theta[i]}\right)^{r},\\
&\qquad (\text{where } \mathit{\Delta}_{\hat{\jmath},\hat{\imath}} := g_1^{-\mathbf{J}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\jmath}])/\mathbf{F}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\jmath}])}(g^{\alpha})^{s_{\hat{\jmath},\hat{\imath}}\mathbf{F}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\imath}])}g^{s_{\hat{\jmath},\hat{\imath}}\mathbf{J}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\jmath}])}),\\
V_{\hat{\jmath},\hat{\imath}} &:= g_1^{-1/\mathbf{F}_{\hat{\jmath},\hat{\imath}}(R_{\hat{\imath},\theta}[\hat{\jmath}])}g^{s_{\hat{\jmath},\hat{\imath}}},\\
V_{j,\hat{\imath}} &:= g^{s_{j,\hat{\imath}}} \quad (\text{for } j\in[\log T_{\hat{\imath}}-1]\setminus\{\hat{\jmath}\}),\\
V_{ji} &:= g^{s_{ji}} \quad (\text{for } i\in[1,D]\setminus\{\hat{\imath}\}, j\in[\log T_i-1]),\\
V'_{ji} &:= g^{t_{ji}} \quad (\text{for } i\in[1,D], j\in[\log T_i-1]),\\
W &:= g^{r}.
\end{aligned}
$$

$\mathcal{B}$ returns $\sigma_\theta := (U, \{V_{ji}, V'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, W)$ to $\mathcal{A}$. Since $V_{\hat{\jmath},\hat{\imath}}$ and $\mathit{\Delta}_{\hat{\jmath},\hat{\imath}}$ distribute identically to $g^{s'}$ and $g_1^{\alpha}(u_{\hat{\jmath},\hat{\imath}}v_0^{R_{\hat{\imath},0}[\hat{\jmath}]})^{s'}$ for $s' \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$ respectively, the signature correctly distributes.

The case S2: $\mathcal{B}$ behaves analogously to the case S1.

The case S3: Obviously $\mathbf{K}(m_\theta)=1$ implies that $\mathbf{F}(m_\theta)\neq 0$. Let $r \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. For every $i\in[1,D]$ and $j\in[0,\log T_i-1]$, let $s_{ji}, t_{ji} \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. $\mathcal{B}$ computes

$$
\begin{aligned}
U &:= \mathit{\Delta}\prod_{i\in[1,D]}\prod_{j\in[0,\log T_i-1]}\left(u_{ji}v_0^{R_{i,\theta}[j]}\right)^{s_{ji}}\left(w_{ji}v_0^{\hat{L}_{i,\theta}[j]}\right)^{t_{ji}},\\
&\qquad (\text{where } \mathit{\Delta} := g_1^{-\mathbf{J}(m_\theta)/\mathbf{F}(m_\theta)}(g^{\alpha})^{r\mathbf{F}(m_\theta)}g^{r\mathbf{J}(m_\theta)}),\\
V_{ji} &:= g^{s_{ji}},\\
V'_{ji} &:= g^{t_{ji}} \quad (\text{for } i\in[1,D], j\in[\log T_i-1]),\\
W &:= g_1^{-1/\mathbf{F}(m_\theta)}g^{r}.
\end{aligned}
$$

$\mathcal{B}$ returns $\sigma_\theta := (U, \{V_{ji}, V'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, W)$ to $\mathcal{A}$. The signature correctly distributes.

The case S4: $\mathcal{B}$ aborts the simulation.

When $\mathcal{A}$ finally outputs a forged signature σ^* for $(m^*, \{L_i^*, R_i^* \mid i\in[1,D]\})$, $\mathcal{B}$ takes different actions in the following two cases:

(F1) $\displaystyle\bigwedge_{i\in[1,D]}\bigwedge_{i\in[0,\log T_i-1]}\mathbf{F}_{ji}(R_i^*[j]) = \mathbf{H}_{ji}(\hat{L}_i^*[j]) = 0 \bigwedge \mathbf{F}(m^*) = 0,$ and

(F2) *Otherwise*,

where $\hat{L}_i^* := T_i - 1 - L_i^*$.

The case F1: If σ^* is a correct signature, it can be described as

$$\sigma^* = (U, \{V_{ji}, V'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, W)$$

$$= \left(\begin{array}{c} g_1^{\alpha} \prod_{i \in [1,D]} \prod_{i \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i^*[j]})^{s_{ji}} (w_{ji} v_0^{\hat{L}_i^*[j]})^{t_{ji}} (u \prod_{i \in [0,N-1]} v_i^{m^*[i]})^r, \\ \{g^{s_{ji}}, g^{t_{ji}} \mid i \in [1,D], j \in [0, \log T_i - 1]\}, g^r \end{array} \right),$$

where $s_{ji}, t_{ji}, r \in \mathbb{Z}_p$. Note that the case F1 implies that

$$\bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} \left[u_{ji} v_0^{R_i^*[j]} = g^{\alpha \mathbf{F}_{ji}(R_i^*[j]) + \mathbf{J}_{ji}(R_i^*[j])} = g^{\mathbf{J}_{ji}(R_i^*[j])} \right]$$
$$\bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} \left[w_{ji} v_0^{\hat{L}_i^*[j]} = g^{\alpha \mathbf{H}_{ji}(\hat{L}_i^*[j]) + \mathbf{Q}_{ji}(\hat{L}_i^*[j])} = g^{\mathbf{Q}_{ji}(\hat{L}_i^*[j])} \right]$$
$$\bigwedge \left[u \prod_{i \in [0,N-1]} v_i^{m^*[i]} = g^{\alpha \mathbf{F}(m^*) + \mathbf{J}(m^*)} = g^{\mathbf{J}(m^*)} \right].$$

$\mathcal{B}$ outputs U/X, where $X := W^{\mathbf{J}(m^*)} \prod_{i \in [1,D]} \prod_{i \in [0, \log T_i - 1]} V_{ji}^{\mathbf{J}_{ji}(R_i^*[j])} V_{ji}'^{\mathbf{Q}_{ji}(\hat{L}_i^*[j])}$, as an answer for the co-CDH problem. If σ^* is a correct signature, the answer is the correct one, i.e., $g_1^{\alpha} = g^{\alpha\beta}$.

<u>The case F2:</u> $\mathcal{B}$ aborts the simulation.

$\mathcal{B}$ behaves as above. Let **Abt** denote the event where $\mathcal{B}$ aborts. For any event $\mathbf{X}$, $\neg\mathbf{X}$ denote negation of the event. We obtain $\mathtt{Adv}_{\mathcal{B}}^{\mathtt{co\text{-}CDH}}(\lambda) = \Pr[g^{\alpha\beta} \leftarrow \mathcal{B} \bigwedge \mathbf{Abt}] + \Pr[g^{\alpha\beta} \leftarrow \mathcal{B} \bigwedge \neg\mathbf{Abt}] \geq \Pr[g^{\alpha\beta} \leftarrow \mathcal{B} \bigwedge \neg\mathbf{Abt}] = \Pr[g^{\alpha\beta} \leftarrow \mathcal{B} \mid \neg\mathbf{Abt}] \Pr[\neg\mathbf{Abt}]$. Since, in any case where $\mathcal{B}$ does not abort the simulation, $\mathcal{B}$ perfectly simulates the existential unforgeability experiment for $\mathcal{A}$, and $\mathcal{B}$ correctly answers if and only if $\mathcal{A}$ makes the experiment output 1, we further obtain

$$\begin{aligned} \mathtt{Adv}_{\mathcal{B}}^{\mathtt{co\text{-}CDH}}(\lambda) &\geq \Pr\left[1 \leftarrow \boldsymbol{Expt}_{\Pi_{\mathrm{SB}},\mathcal{A}}^{\mathtt{EUF\text{-}CMA}}(1^{\lambda}, D, T_1, \cdots, T_D)\right] \Pr\left[\neg\mathbf{Abt}\right] \\ &= \mathtt{Adv}_{\Pi_{\mathrm{SB}},\mathcal{A},D,T_1,\cdots,T_D}^{\mathtt{EUF\text{-}CMA}}(\lambda) \cdot \Pr\left[\neg\mathbf{Abt}\right]. \end{aligned} \tag{6}$$

$\Pr[\neg\mathbf{Abt}]$ can be analysed as follows.

$$\begin{aligned} &\Pr[\neg\mathbf{Abt}] \\ &= \Pr[\neg\mathbf{Abt}_F \mid \neg\mathbf{Abt}_H] \Pr[\neg\mathbf{Abt}_H] && (7) \\ &= \Pr[\neg\mathbf{Abt}_H \mid \neg\mathbf{Abt}_F] \Pr[\neg\mathbf{Abt}_F] \\ &= (1 - \Pr[\mathbf{Abt}_H \mid \neg\mathbf{Abt}_F]) \Pr[\neg\mathbf{Abt}_F] \\ &= \left(1 - \Pr\left[\bigvee_{\iota \in [1,q_r]} \mathbf{Abt}_R^{\iota} \bigvee_{\theta \in [1,q_s]} \mathbf{Abt}_S^{\theta} \middle| \neg\mathbf{Abt}_F \right]\right) \Pr[\neg\mathbf{Abt}_F] && (8) \\ &\geq \left(1 - \sum_{\iota \in [1,q_r]} \Pr[\mathbf{Abt}_R^{\iota} \mid \neg\mathbf{Abt}_F] - \sum_{\theta \in [1,q_s]} \Pr[\mathbf{Abt}_S^{\theta} \mid \neg\mathbf{Abt}_F] \right) \Pr[\neg\mathbf{Abt}_F] \\ & && (9) \end{aligned}$$

$$\geq \left(1 - \frac{1}{n}\left(\left(\sum_{i\in[1,D]} \log T_i\right) q_r + q_s\right)\right) \frac{1}{\{n(N+1)\}^{2\left(\sum_{i\in[1,D]} \log T_i\right)+1}} \tag{10}$$

$$= \frac{1}{2\left\{2\left(\left(\sum_{i\in[1,D]} \log T_i\right) q_r + q_s\right)(N+1)\right\}^{2\left(\sum_{i\in[1,D]} \log T_i\right)+1}}. \tag{11}$$

For (7), $\mathbf{Abt}_H$ denotes the event where $\mathcal{B}$ has already aborted right before $\mathcal{A}$ outputs the forged signature, and $\mathbf{Abt}_F$ denotes the event where $\mathcal{B}$ aborts after $\mathcal{A}$ outputs the forged signature. For (8), $\mathbf{Abt}_R^{\iota}$ (resp. $\mathbf{Abt}_S^{\theta}$) denotes the event where $\mathcal{B}$ aborts on the ι-th $\mathfrak{Reveal}$ query (resp. the θ-th $\mathfrak{Sign}$ query). (10) is obtained because of Lemma 1, Lemma 2 (which is a sub-lemma of Lemma 1), Lemma 3 and Lemma 4. (11) is because of $n := 2((\sum_{i\in[1,D]} \log T_i)q_r + q_s)$. Finally, based on (6) and (11), we obtain (1).

Let us present a conciser result. Let us consider a case where (i) every threshold value $d_\iota \in [1, D]$ queried to $\mathfrak{Reveal}$ is D and (ii) there exists $T \in \mathbb{T}$ s.t. for every $i \in [1, D]$, $T_i = T$. Applying the same lemmata to (9), we obtain

$$\begin{aligned}\Pr[\neg\mathbf{Abt}] &\geq (1 - \frac{1}{n}(\log T \cdot q_r + q_s)) \frac{1}{\{n(N+1)\}^{2D\log T+1}} \\ &= \frac{1}{2\{2(\log T \cdot q_r + q_s)(N+1)\}^{2D\log T+1}}.\end{aligned}$$

Thus, we obtain (2). □

Lemma 1. *$\forall\iota \in [1, q_r]$, $\Pr[\mathbf{Abt}_R^{\iota} \mid \neg\mathbf{Abt}_F] \leq \sum_{i\in[1,D]} \log T_i/n$. If $\bigwedge_{i\in[1,D]} T_i = T \bigwedge_{\iota\in[1,q_r]} d_\iota = D$, then $\forall\iota \in [1, q_r]$, $\Pr[\mathbf{Abt}_R^{\iota} \mid \neg\mathbf{Abt}_F] \leq \log T/n$.*

Proof. For $\iota \in [1, q_r]$ and $i \in [1, D]$, we define $\mathbf{A}_{i,\iota}$ as an event where the condition in (3) is satisfied. The event and its negation are formally described as follows.

$$\mathbf{A}_{i,\iota} := \left[\begin{array}{l} \bigvee_{\substack{j\in[0,\log T_i-1] \\ \text{s.t. } r_{i,\iota}[j]=1}} \left[\mathbf{K}_{ji}(1) = 1 \bigwedge \left[j \neq 0 \implies \bigwedge_{\substack{k\in[0,j-1] \\ \text{s.t. } r_{i,\iota}[k]=0}} \mathbf{K}_{ki}(1) = 1\right]\right] \\ \bigvee_{\substack{j\in[0,\log T_i-1] \\ \text{s.t. } \hat{l}_{i,\iota}[j]=1}} \left[\mathbf{U}_{ji}(1) = 1 \bigwedge \left[j \neq 0 \implies \bigwedge_{\substack{k\in[0,j-1] \\ \text{s.t. } \hat{l}_{i,\iota}[k]=0}} \mathbf{U}_{ki}(1) = 1\right]\right] \end{array}\right].$$

$$\neg\mathbf{A}_{i,\iota} := \left[\begin{array}{l} \bigwedge_{\substack{j\in[0,\log T_i-1] \\ \text{s.t. } r_{i,\iota}[j]=1}} \left[\mathbf{K}_{ji}(1) = 0 \bigvee \left[j \neq 0 \implies \bigvee_{\substack{k\in[0,j-1] \\ \text{s.t. } r_{i,\iota}[k]=0}} \mathbf{K}_{ki}(1) = 0\right]\right] \\ \bigwedge_{\substack{j\in[0,\log T_i-1] \\ \text{s.t. } \hat{l}_{i,\iota}[j]=1}} \left[\mathbf{U}_{ji}(1) = 0 \bigvee \left[j \neq 0 \implies \bigvee_{\substack{k\in[0,j-1] \\ \text{s.t. } \hat{l}_{i,\iota}[k]=0}} \mathbf{U}_{ki}(1) = 0\right]\right] \end{array}\right].$$

Upper bound of $\Pr\left[\mathbf{Abt}_R^{\iota} \mid \neg\mathbf{Abt}_F\right]$ is derived as follows.

$$\Pr\left[\mathbf{Abt}_R^{\iota} \mid \neg\mathbf{Abt}_F\right]$$

$$
\begin{aligned}
&= \Pr\left[|\mathbb{I}_\iota| < D - d_\iota + 1 \mid \neg\mathbf{Abt}_F\right] && (12)\\
&\leq \Pr\left[|\mathbb{I}_\iota \cap \mathbb{I}'_\iota| < D - d_\iota + 1 \mid \neg\mathbf{Abt}_F\right] && (13)\\
&\leq \max_{\mathbb{I}^*_\iota \subseteq \mathbb{I}'_\iota \text{ s.t. } |\mathbb{I}^*_\iota| = D - d_\iota + 1} \left\{\Pr\left[|\mathbb{I}_\iota \cap \mathbb{I}^*_\iota| < D - d_\iota + 1 \mid \neg\mathbf{Abt}_F\right]\right\} && (14)\\
&= \max_{\mathbb{I}^*_\iota \subseteq \mathbb{I}'_\iota \text{ s.t. } |\mathbb{I}^*_\iota| = D - d_\iota + 1} \left\{\Pr\left[\bigvee_{i \in \mathbb{I}^*_\iota} \neg\mathbf{A}_{i,\iota} \;\middle|\; \neg\mathbf{Abt}_F\right]\right\} && (15)\\
&\leq \max_{\mathbb{I}^*_\iota \subseteq \mathbb{I}'_\iota \text{ s.t. } |\mathbb{I}^*_\iota| = D - d_\iota + 1} \left\{\sum_{i \in \mathbb{I}^*_\iota} \Pr\left[\neg\mathbf{A}_{i,\iota} \mid \neg\mathbf{Abt}_F\right]\right\}\\
&\leq \max_{\mathbb{I}^*_\iota \subseteq \mathbb{I}'_\iota \text{ s.t. } |\mathbb{I}^*_\iota| = D - d_\iota + 1} \left\{\sum_{i \in \mathbb{I}^*_\iota} \frac{\log T_i}{n}\right\} && (16)\\
&= \frac{\sum_{i \in [1, D - d_\iota + 1]} \log T'_i}{n} && (17)\\
&\leq \frac{\sum_{i \in [1, D - d_\iota + 1]} \log T^*_i}{n} && (18)
\end{aligned}
$$

(12) is obtained because of definition of the event $\mathbf{Abt}^\iota_R$. $\mathbb{I}_\iota$ is a set of $i \in [1, D]$ such that the event $\mathbf{A}_{i,\iota}$ occurs. Let $\mathbb{I}'_\iota$ denote a set of $i \in [1, D]$ s.t. $\neg[L^*_i \leq l_{i,\iota} \leq r_{i,\iota} \leq R^*_i]$. Because of definition of existential unforgeability, it holds that $|\mathbb{I}'_\iota| \geq D - d_\iota + 1$. (13) obviously holds since $\mathbb{I}_\iota \cap \mathbb{I}'_\iota$ is a subset of $\mathbb{I}_\iota$. (14) holds because for every $\mathbb{I}_1, \mathbb{I}_2$ s.t. $\mathbb{I}_1 \subset \mathbb{I}_2 \subseteq \mathbb{I}'_\iota$, it holds that $\Pr\left[|\mathbb{I}_\iota \cap \mathbb{I}_1| < D - d_\iota + 1 \mid \neg\mathbf{Abt}_F\right] > \Pr\left[|\mathbb{I}_\iota \cap \mathbb{I}_2| < D - d_\iota + 1 \mid \neg\mathbf{Abt}_F\right]$. (15) is because for every $\mathbb{I}^*_\iota \subseteq \mathbb{I}'_\iota$ s.t. $|\mathbb{I}^*_\iota| = D - d_\iota + 1$, $\Pr[|\mathbb{I}_\iota \cap \mathbb{I}^*_\iota| < D - d_\iota + 1 \mid \neg\mathbf{Abt}_F] = \Pr[\vee_{i \in \mathbb{I}^*_\iota} \neg\mathbf{A}_{i,\iota} \mid \neg\mathbf{Abt}_F]$. (16) is because of Lemma 2. In (17), for $i \in [1, D - d_\iota + 1]$, T'_i is the i-th largest integer among $\{T_k \mid k \in \mathbb{I}'_\iota\}$. Formally, for every $i \in [1, D - d_\iota + 1]$, $T'_i := \max\left\{\{T_k \mid k \in \mathbb{I}'_\iota\} \setminus_{j \in [1, i-1]} \left\{T'_j\right\}\right\}$. In (18), for $i \in [1, D - d_\iota + 1]$, T^*_i is the i-th largest integer among $\{T_1, \cdots, T_D\}$. Formally, for every $i \in [1, D - d_\iota + 1]$, $T^*_i := \max\left\{\{T_1, \cdots, T_D\} \setminus_{j \in [1, i-1]} \left\{T^*_j\right\}\right\}$.

When $d_\iota = 1$, (18) is equal to $\frac{\sum_{i \in [1, D]} \log T_i}{n}$. When $d_\iota = D$ and $\wedge^D_{i=1} T_i = T$, (18) is equal to $\frac{\log T}{n}$. □

Lemma 2. *$\forall \iota \in [1, q_r]$, $\forall i \in [1, D]$ s.t. $\neg[L^*_i \leq l_{i,\iota} \leq r_{i,\iota} \leq R^*_i]$, $\Pr\left[\neg\mathbf{A}_{i,\iota} \mid \neg\mathbf{Abt}_F\right] \leq \log T_i / n$.*

Proof. Let us consider the following two cases: (I) $r_{i,\iota} > R^*_i$ and (II) $l_{i,\iota} < L^*_i$.

Let us analyse the probability in the first case. We obtain

$$
\begin{aligned}
&\Pr[\neg\mathbf{A}_{\iota,i} \mid \neg\mathbf{Abt}_F]\\
&= \Pr\left[\bigwedge_{\substack{j \in [0, \log T_i - 1] \\ \text{s.t. } r_{\iota,i}[j] = 1}} \left[\mathbf{K}_{j,i}(1) = 0 \bigvee \left[j \neq 0 \implies \bigvee_{\substack{k \in [0, j-1] \\ \text{s.t. } r_{i,\iota}[k] = 0}} \mathbf{K}_{k,i}(1) = 0\right]\right] \;\middle|\; \neg\mathbf{Abt}_F\right]
\end{aligned}
$$

$$\leq \Pr\left[\mathbf{K}_{k_i,i}(1) = 0 \bigvee \left[k_i \neq 0 \implies \bigvee_{\substack{k\in[0,k_i-1]\\ \text{s.t. } r_{i,\iota}[k]=0}} \mathbf{K}_{k,i}(1) = 0\right] \middle| \neg\mathbf{Abt}_F\right] \tag{19}$$

$$= \Pr\left[\mathbf{K}_{k_i,i}(1) = 0 \bigvee_{\substack{k\in[0,k_i-1]\\ \text{s.t. } r_{i,\iota}[k]=0}} \mathbf{K}_{k,i}(1) = 0 \middle| \neg\mathbf{Abt}_F\right]$$

$$\leq \Pr\left[\mathbf{K}_{k_i,i}(1) = 0 \mid \neg\mathbf{Abt}_F\right] + \sum_{\substack{k\in[0,k_i-1]\\ \text{s.t. } r_{i,\iota}[k]=0}} \Pr\left[\mathbf{K}_{k,i}(1) = 0 \mid \neg\mathbf{Abt}_F\right] \tag{20}$$

(19) holds since it is true that $r_{i,\iota} > R_i^*$ implies that there exists $k_i \in [0, \log T_i - 1]$ s.t. $[k_i \neq 0 \implies \wedge_{j\in[0,k_i-1]} r_{i,\iota}[j] = R_i^*[j]]$ and $[r_{i,\iota}[k_i] = 1 \wedge R_i^*[k_i] = 0]$.

We obtain

$$\Pr\left[\mathbf{K}_{k_i,i}(1) = 0 \mid \neg\mathbf{Abt}_F\right]$$

$$= \Pr\left[\mathbf{L}_{k_i,i}(1) = 0 \middle| \begin{array}{c} \bigwedge_{\tau\in[1,D]} \bigwedge_{j\in[0,\log T_\tau-1]} \mathbf{L}_{j,\tau}(R_\tau^*[j]) = \mathbf{R}_{j,\tau}(\hat{L}_\tau^*[j]) = 0 \\ \bigwedge \mathbf{L}(m^*) = 0 \end{array}\right] \tag{21}$$

$$= \Pr\left[\mathbf{L}_{k_i,i}(1) = 0 \mid \mathbf{L}_{k_i,i}(R_i^*[k_i]) = 0\right] \tag{22}$$

$$= \Pr[x_{k_i,i} + y_0 = 0 \mod n \mid x_{k_i,i} = 0 \mod n] \tag{23}$$

$$= 1/n \tag{24}$$

(21) obviously holds. (22) holds since the event $\mathbf{L}_{k_i,i}(1) = 0$ occurs independently from any of the following events

- $\mathbf{L}_{j,\tau}(R_\tau^*[j]) = 0$ (for $\tau \in [1, D] \setminus \{i\}$ and $j \in [0, \log T_\tau - 1]$),
- $\mathbf{L}_{j,i}(R_i^*[j]) = 0$ (for $j \in [0, \log T_\tau - 1] \setminus \{k_i\}$),
- $\mathbf{R}_{j,\tau}(\hat{L}_\tau^*[j]) = 0$ (for $\tau \in [1, D]$ and $j \in [0, \log T_\tau - 1]$),
- $\mathbf{L}(m^*) = 0$.

(23) holds since $R_i^*[k_i] = 0$. (24) holds since y_0 is chosen uniformly at random from $\mathbb{Z}_n$. By the same argument, for any $k \in [0, k_i - 1]$ s.t. $r_{i,\iota}[k] = 0$, we obtain

$$\Pr\left[\mathbf{K}_{k,i}(1) = 0 \mid \neg\mathbf{Abt}_F\right]$$

$$= \Pr\left[\mathbf{L}_{k,i}(1) = 0 \mid \mathbf{L}_{k,i}(R_i^*[k]) = 0\right] \tag{25}$$

$$= \Pr[x_{k,i} + y_0 = 0 \mod n \mid x_{k,i} = 0 \mod n] \tag{26}$$

$$= 1/n \tag{27}$$

(26) holds since $r_{i,\iota}[j] = R_i^*[j]$ for every $j \in [0, k_i - 1]$ because of definition of k_i. By (20), (24) and (27), we obtain $\Pr[\neg\mathbf{A}_{\iota,i} \mid \neg\mathbf{Abt}_F] \leq \frac{1}{n} + \sum_{k\in[0,k_i-1] \text{ s.t. } r_{i,\iota}[k]=0} \frac{1}{n} \leq \frac{\log T_i}{n}$. For the case (II), by using the same argument, we obtain $\Pr\left[\neg\mathbf{A}_{i,\iota} \mid \neg\mathbf{Abt}_F\right] \leq \frac{\log T_i}{n}$. □

Lemma 3. *$\forall\theta \in [1, q_s]$, $\Pr[\mathbf{Abt}_S^\theta \mid \neg\mathbf{Abt}_F] \leq 1/n$.*

Proof. For every $\theta \in [1, q_s]$, $\mathcal{A}$ must query $\{L_{i,\theta}, R_{i,\theta} \mid i \in [1, D]\}$ and m_θ s.t. $(\{L_{i,\theta}, R_{i,\theta} \mid i \in [1, D]\}, m_\theta) \neq (\{L_i^*, R_i^* \mid i \in [1, D]\}, m^*)$, which implies that at least one of the following 3 conditions holds: (I) $\exists \hat{i} \in [1, D]$, $\exists \hat{j} \in [0, \log T_i - 1]$, $R_{\hat{i},\theta}[\hat{j}] \neq R_i^*[\hat{j}]$, (II) $\exists \hat{i} \in [1, D]$, $\exists \hat{j} \in [0, \log T_{\hat{i}} - 1] L_{\hat{i},\theta}[\hat{j}] \neq L_{\hat{i}}^*[\hat{j}]$ and (III) $\exists \hat{j} \in [0, N - 1]$, $m_\theta[\hat{j}] \neq m^*[\hat{j}]$.

For the case where the condition (I) holds, we obtain

$$
\begin{aligned}
&\Pr[\mathbf{Abt}_S^\theta \mid \neg \mathbf{Abt}_F] = \Pr[\bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} \mathbf{K}_{j,i}(R_{i,\theta}[j]) = 0 \mid \neg \mathbf{Abt}_F] \\
&\leq \Pr[\mathbf{K}_{\hat{j},\hat{i}}(R_{\hat{i},\theta}[\hat{j}]) = 0 \mid \neg \mathbf{Abt}_F] \\
&= \Pr\left[\mathbf{L}_{\hat{j},\hat{i}}(R_{\hat{i},\theta}[\hat{j}]) = 0 \middle| \begin{array}{r} \bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} \mathbf{L}_{j,i}(R_i^*[j]) = \mathbf{R}_{j,i}(\hat{L}_i^*[j]) = 0 \\ \bigwedge \mathbf{L}(m^*) = 0 \end{array} \right] \qquad (28) \\
&= \Pr\left[\mathbf{L}_{\hat{j},\hat{i}}(R_{\hat{i},\theta}[\hat{j}]) = 0 \middle| \mathbf{L}_{\hat{j},\hat{i}}(R_{\hat{i}}^*[\hat{j}]) = 0 \right] \\
&= \Pr\left[x_{\hat{j},\hat{i}} + y_0 \cdot R_{\hat{i},\theta}[\hat{j}] = 0 \mod n \middle| x_{\hat{j},\hat{i}} + y_0 \cdot R_{\hat{i}}^*[\hat{j}] = 0 \mod n \right] \qquad (29) \\
&= 1/n. \qquad (30)
\end{aligned}
$$

For (28), $\hat{L}_i^*$ is $T_i - 1 - L_i^*$. The transformation from (29) to (30) holds in either of the following 2 cases, namely (i) $R_{\hat{i},\theta}[\hat{j}] = 0$ and $R_{\hat{i}}^*[\hat{j}] = 1$, and (ii) $R_{\hat{i},\theta}[\hat{j}] = 1$ and $R_{\hat{i}}^*[\hat{j}] = 0$, because of the fact that $x_{\hat{j},\hat{i}}, y_0 \overset{\mathrm{U}}{\leftarrow} \mathbb{Z}_p$.

For the case (II), by using the same argument, we obtain $\Pr[\mathbf{Abt}_S^\theta \mid \neg \mathbf{Abt}_F] \leq 1/n$.

For the case (III), we obtain $\Pr[\mathbf{Abt}_S^\theta \mid \neg \mathbf{Abt}_F] = \Pr[\mathbf{K}(m_\theta) = 0 \mid \neg \mathbf{Abt}_F] = \Pr[\mathbf{L}(m_\theta) = 0 \mid \bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} \mathbf{L}_{j,i}(R_i^*[j]) = \mathbf{R}_{j,i}(\hat{L}_i^*[j]) = 0 \bigwedge \mathbf{L}(m^*) = 0] = \Pr[\mathbf{L}(m_\theta) = 0 \mid \mathbf{L}(m^*) = 0] = 1/n$. □

Lemma 4. $\Pr[\neg \mathbf{Abt}_F] \geq \frac{1}{\{n(N+1)\}^{2(\sum_{i \in [1,D]} \log T_i) + 1}}$.

Proof. We obtain

$$
\begin{aligned}
&\Pr[\neg \mathbf{Abt}_F] \\
&= \Pr\left[\begin{array}{c} \bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} [x_{ji} + R_i^*[j] \cdot y_0 = n \cdot k_{ji}] \\ \bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i - 1]} [z_{ji} + L_i^*[j] \cdot y_0 = n \cdot s_{ji}] \bigwedge \left[\hat{x} + \sum_{j \in [0, N-1]} m^*[j] \cdot y_j = n \cdot \hat{k} \right] \end{array} \right]
\end{aligned}
$$

$$
= \Pr\left[\begin{array}{c}
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \bigvee_{k'_{ji}\in[0,N]} \left[x_{ji} + R_i^*[j]\cdot y_0 = n\cdot k'_{ji} \wedge k'_{ji} = k_{ji}\right] \\
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \bigvee_{s'_{ji}\in[0,N]} \left[z_{ji} + L_i^*[j]\cdot y_0 = n\cdot s'_{ji} \wedge s'_{ji} = s_{ji}\right] \\
\bigwedge \bigvee_{k'\in[0,N]} \left[\hat{x} + \sum_{j\in[0,N-1]} m^*[j]\cdot y_j = n\cdot k' \wedge k' = \hat{k}\right]
\end{array}\right]
$$

$$
= \Pr\left[\begin{array}{c}
\bigwedge_{i\in[1,D]} \bigwedge_{i\in[0,\log T_i]} \bigvee_{k'_{ji}\in[0,N]} \left[\mathbf{X}_{j,i,k'_{ji}} \wedge \tilde{\mathbf{X}}_{j,i,k'_{ji}}\right] \\
\bigwedge_{i\in[1,D]} \bigwedge_{i\in[0,\log T_i-1]} \bigvee_{s'_{ji}\in[0,N]} \left[\mathbf{Y}_{j,i,s'_{ji}} \wedge \tilde{\mathbf{Y}}_{j,i,s'_{ji}}\right] \bigwedge \bigvee_{k'\in[0,N]} \left[\mathbf{Z}_{k'} \wedge \tilde{\mathbf{Z}}_{k'}\right]
\end{array}\right] \tag{31}
$$

$$
= \Pr\left[\begin{array}{c}
\bigvee_{\substack{\left(\left\{k'_{ji},s'_{ji} \mid i\in[1,D],j\in[0,\log T_i-1]\right\},k'\right) \\ \in[0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}}} \left\{ \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i]} \left[\mathbf{X}_{j,i,k'_{ji}} \wedge \tilde{\mathbf{X}}_{j,i,k'_{ji}}\right] \right. \\
\left. \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \left[\mathbf{Y}_{j,i,s'_{ji}} \wedge \tilde{\mathbf{Y}}_{j,i,s'_{ji}}\right] \bigwedge \left[\mathbf{Z}_{k'} \wedge \tilde{\mathbf{Z}}_{k'}\right] \right\}
\end{array}\right]
$$

$$
= \sum_{\substack{\left(\left\{k'_{ji},s'_{ji} \mid i\in[1,D],j\in[0,\log T_i-1]\right\},k'\right) \\ \in[0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}}} \Pr\left[\begin{array}{c}
\bigwedge_{i\in[1,D]} \bigwedge_{i\in[0,\log T_i]} \left[\mathbf{X}_{j,i,k'_{ji}} \wedge \tilde{\mathbf{X}}_{j,i,k'_{ji}}\right] \\
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \left[\mathbf{Y}_{j,i,s'_{ji}} \wedge \tilde{\mathbf{Y}}_{j,i,s'_{ji}}\right] \\
\bigwedge \left[\mathbf{Z}_{k'} \wedge \tilde{\mathbf{Z}}_{k'}\right]
\end{array}\right] \tag{32}
$$

$$
\begin{aligned}
&= \sum_{\substack{\left(\left\{k'_{ji},s'_{ji} \mid i\in[1,D],j\in[0,\log T_i-1]\right\},k'\right) \\ \in[0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}}} \left\{ \Pr\left[\begin{array}{c}
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{X}_{j,i,k'_{ji}} \\
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{Y}_{j,i,s'_{ji}} \bigwedge \mathbf{Z}_{k'}
\end{array}\right] \right. \\
&\quad \left. \cdot \Pr\left[\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \tilde{\mathbf{X}}_{j,i,k'_{ji}} \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \tilde{\mathbf{Y}}_{j,i,s'_{ji}} \bigwedge \tilde{\mathbf{Z}}_{k'} \right] \right\}
\end{aligned} \tag{33}
$$

$$
\begin{aligned}
&= \frac{1}{(N+1)^{2\sum_{i\in[1,D]}(\log T_i)+1}} \\
&\quad \sum_{\substack{\left(\left\{k_{j,i},s_{j,i} \mid i\in[1,D],j\in[0,\log T_i-1]\right\},\hat{k}\right) \\ \in[0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}}} \Pr\left[\begin{array}{c}
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{X}_{j,i,k_{ji}} \\
\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{Y}_{j,i,s_{ji}} \bigwedge \mathbf{Z}_{\hat{k}}
\end{array}\right]
\end{aligned}
$$

$$
\begin{aligned}
&= \frac{1}{(N+1)^{2\sum_{i\in[1,D]}(\log T_i)+1}} \\
&\quad \cdot \Pr\left[\bigvee_{\substack{(\{k'_{ji},s'_{ji} | i\in[1,D], j\in[0,\log T_i-1]\},k')\\ \in[0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}}} \left[\begin{array}{c} \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{X}_{j,i,k_{ji}} \\ \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{Y}_{j,i,s_{ji}} \bigwedge \mathbf{Z}_{\hat{k}} \end{array} \right] \right] \qquad (34) \\
&= \frac{1}{(N+1)^{2\sum_{i\in[1,D]}(\log T_i)+1}} \Pr\left[\begin{array}{c} \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \bigvee_{k'_{ji}\in[0,N]} \mathbf{X}_{j,i,k'_{ji}} \\ \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \bigvee_{s'_{ji}\in[0,N]} \mathbf{Y}_{j,i,s'_{ji}} \bigwedge \bigvee_{k'\in[0,N]} \mathbf{Z}_{k'} \end{array} \right] \\
&= \frac{1}{(N+1)^{2\sum_{i\in[1,D]}(\log T_i)+1}} \Pr\left[\begin{array}{c} \bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i-1]} \mathbf{L}_{ji}(R_i^*[j]) = \mathbf{R}_{ji}(L_i^*[j]) = 0 \\ \bigwedge \mathbf{L}(m^*) = 0 \end{array} \right] \\
&= \frac{1}{\{n\,(N+1)\}^{2\sum_{i\in[1,D]}(\log T_i)+1}}.
\end{aligned}
$$

For (31), we used

$$
\begin{aligned}
&\mathbf{X}_{j,i,k'_{ji}} := \left[x_{ji} + R_i^*[j]\cdot y_0 = n\cdot k'_{ji}\right], \tilde{\mathbf{X}}_{j,i,k'_{ji}} := \left[k'_{ji} = k_{ji}\right], \\
&\mathbf{Y}_{j,i,s'_{ji}} := \left[z_{ji} + L_i^*[j]\cdot y_0 = n\cdot s'_{ji}\right], \tilde{\mathbf{Y}}_{j,i,s'_{ji}} := \left[s'_{ji} = s_{ji}\right], \\
&\mathbf{Z}_{k'} := \left[\hat{x} + \sum_{j\in[0,N-1]} m^*[j]\cdot y_j = n\cdot k'\right], \tilde{\mathbf{Z}}_{k'} := \left[k' = \hat{k}\right].
\end{aligned}
$$

The transformation to (32) is correct since for every $(\{k'_{ji}, s'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, k') \in [0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}$ and every $(\{k^*_{ji}, s^*_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, k^*) \in [0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}$ s.t. $(\{k'_{ji}, s'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, k') \neq (\{k^*_{ji}, s^*_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, k^*)$, the following 2 events are exclusive.

1. $\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i]} [\mathbf{X}_{j,i,k'_{ji}} \wedge \tilde{\mathbf{X}}_{j,i,k'_{ji}} \mathbf{Y}_{j,i,s'_{ji}} \wedge \tilde{\mathbf{Y}}_{j,i,s'_{ji}}] \bigwedge [\mathbf{Z}_{k'} \wedge \tilde{\mathbf{Z}}_{k'}]$.
2. $\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i]} [\mathbf{X}_{j,i,k^*_{ji}} \wedge \tilde{\mathbf{X}}_{j,i,k^*_{ji}} \mathbf{Y}_{j,i,s^*_{ji}} \wedge \tilde{\mathbf{Y}}_{j,i,s^*_{ji}}] \bigwedge [\mathbf{Z}_{k^*} \wedge \tilde{\mathbf{Z}}_{k^*}]$.

The transformation to (33) is correct since for every $(\{k'_{ji}, s'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, k') \in [0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}$, the following 2 events are independent.

1. $\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i]} [\mathbf{X}_{j,i,k'_{ji}} \wedge \mathbf{Y}_{j,i,s'_{ji}}] \bigwedge [\mathbf{Z}_{k'}]$.
2. $\bigwedge_{i\in[1,D]} \bigwedge_{j\in[0,\log T_i]} [\tilde{\mathbf{X}}_{j,i,k'_{ji}} \wedge \tilde{\mathbf{Y}}_{j,i,s'_{ji}}] \bigwedge [\tilde{\mathbf{Z}}_{k'}]$.

The transformation to (34) is correct since for every $(\{k'_{ji}, s'_{ji} \mid i\in[1,D], j\in[0,\log T_i-1]\}, k') \in [0,N]^{2\sum_{i\in[1,D]}(\log T_i)+1}$ and every $(\{k^*_{ji}, s^*_{ji} \mid i\in[1,D], j\in$

$[0, \log T_i - 1]\}, k^*) \in [0, N]^{2 \sum_{i \in [1,D]} (\log T_i) + 1}$ s.t. $(\{k'_{ji}, s'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, k') \neq (\{k^*_{ji}, s^*_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, k^*)$, the following 2 events are exclusive.

1. $\bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i]} [\mathbf{X}_{j,i,k'_{ji}} \wedge \mathbf{Y}_{j,i,s'_{ji}}] \bigwedge [\mathbf{Z}_{k'}]$.
2. $\bigwedge_{i \in [1,D]} \bigwedge_{j \in [0, \log T_i]} [\mathbf{X}_{j,i,k^*_{ji}} \wedge \mathbf{Y}_{j,i,s^*_{ji}}] \bigwedge [\mathbf{Z}_{k^*}]$.

□

3.4 Perfect Privacy of Our MDSBRS Scheme

Theorem 2. *Our MDSBRS scheme Π_{SB} is perfectly private.*

Proof. Among the 4 simulating algorithms $(\widehat{\texttt{Setup}}, \widehat{\texttt{Setup}}, \widehat{\texttt{KDel}}, \widehat{\texttt{Sig}})$ used in $\boldsymbol{Expt}^{\mathsf{PP}}_{\Pi_{\mathrm{SB}}, \mathcal{A}, 1}$, the first 3 algorithms are identical to the original ones of Π_{SB}. $\widehat{\texttt{Sig}}$ directly generates a signature on m under $\{L_i, R_i \mid i \in [1, D]\}$ from msk. Formally, it takes $(msk, m \in \{0,1\}^N, \{L_i, R_i \mid i \in [1, D]\})$, then generates σ as follows.

- For every $i \in [1, D]$, $\hat{L}_i := T_i - 1 - L_i$.
- For every $i \in [1, D]$ and $j \in [0, \log T_i - 1]$, $s_{ji}, t_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $r \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $\sigma := (g_1^{\alpha} \prod_{i \in [1,D]} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s_{ji}} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji}} (u \prod_{j \in [0, N-1]} v_j^{m[j]})^r, \{g^{s_{ji}}, g^{t_{ji}} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, g^r)$.

We prove that every signature σ generated on $\mathfrak{Sign}$ in $\boldsymbol{Expt}^{\mathsf{PP}}_{\Pi_{\mathrm{SB}}, \mathcal{A}, 0}$ distributes identically to one in $\boldsymbol{Expt}^{\mathsf{PP}}_{\Pi_{\mathrm{SB}}, \mathcal{A}, 1}$.

Since the secret-key sk_ι used to generate a signature σ on $\mathfrak{Sign}$ has been honestly generated by `KGen` or `KDel`, it is parsed as $(\{sk_{l_{i,\iota}}, sk_{r_{i,\iota}} \mid i \in [1, D]\})$, and the elements satisfy that there exist $\{A_j \mid j \in [1, d_\iota - 1]\}$, $\{s_{ji}, t_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}$, $\{s'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]$ s.t. $r_{i,\iota}[j] = 0\}$, $\{t'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]$ s.t. $\hat{l}_{i,\iota}[j] = 0\}$ and $\{\delta_i \in [1, D]\}$ such that for every $i \in [1, D]$,

$$- \ sk_{r_i} := \begin{pmatrix} g_1^{\sum_{j \in [1, d_\iota - 1]} A_j i^j + \alpha} g^{\delta_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{r_{i,\iota}[j]})^{s_{ji}}, g^{s_{0,i}}, \cdots, g^{s_{\log T_i - 1, i}}, \\ \left\{ \begin{matrix} g_1^{\sum_{j \in [0, d_\iota - 1]} A_j i^j + \alpha} g^{\delta_i} \prod_{j \in [0, k-1]} (u_{ji} v_0^{r_{i,\iota}[j]})^{s_{ji}} (u_{ki} v_0)^{s'_{ki}}, g^{s'_{ki}} \\ \mid k \in [0, \log T_i - 1] \text{ s.t. } r_i[k] = 0 \end{matrix} \right\} \end{pmatrix},$$

$$- \ sk_{l_i} := \begin{pmatrix} g^{-\delta_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{l}_{i,\iota}[j]})^{t_{ji}}, g^{t_{0,i}}, \cdots, g^{t_{\log T_i - 1, i}}, \\ \left\{ g^{-\delta_i} \prod_{j \in [0, k-1]} (w_{ji} v_0^{\hat{l}_{i,\iota}[j]})^{t_{ji}} (w_{ki} v_0)^{t'_{ki}}, g^{t'_{ki}} \middle| \begin{matrix} k \in [0, \log T_i - 1] \\ \text{s.t. } \hat{l}_i[k] = 0 \end{matrix} \right\} \end{pmatrix}.$$

We firstly derive a set $\mathbb{I} = \{i \in [1, D] \text{ s.t. } L_i \leq l_{i,\iota} \leq r_{i,\iota} \leq R_i\}$ whose cardinality is more than $d-1$. For each $i \in \mathbb{I}$, let $(l'_i, r'_i) := (L_i, R_i)$. Note that $r_{i,\iota} \leq R_i$ guarantees that there exists $k_i \in [0, \log T_i]$ which satisfies $[k_i \neq 0 \implies \bigwedge_{j \in [0, k_i - 1]} r_{i,\iota}[j] = R_i[j]] \bigwedge [k_i \neq \log T_i \implies r_{i,\iota}[k_i] = 0 \bigwedge R_i[k_i] = 1]$. Also note that $l_{i,\iota} \geq L_i$ guarantees that there exists $k'_i \in [0, \log T_i]$ which satisfies $[k'_i \neq 0 \implies \bigwedge_{j \in [0, k'_i - 1]} \hat{l}_{i,\iota}[j] = \hat{L}_i[j]] \bigwedge [k'_i \neq \log T_i \implies \hat{l}_{i,\iota}[k'_i] = 0 \bigwedge \hat{L}_i[k'_i] = 1]$. For each $i \in [1, D] \setminus \mathbb{I}$, let $(l'_i, r'_i) := (l_{i,\iota}, r_{i,\iota})$. We transform sk_ι into $\tilde{sk} \leftarrow \mathtt{KDel}(sk_\iota, \{l'_i, r'_i \mid i \in [1, D]\})$. Let us parse $\tilde{sk}$ as $(\{sk_{l'_i}, sk_{r'_i} \mid i \in [1, D]\})$. We are interested in $\{sk_{l'_i}, sk_{r'_i} \mid i \in \mathbb{I}\}$. They consist of $\{(E_{\log T_i, i}, e_{0,i}, \cdots, e_{\log T_i - 1, i}, \cdots), (D_{\log T_i, i}, d_{0,i}, \cdots, d_{\log T_i - 1, i}, \cdots) \mid i \in \mathbb{I}\}$ and satisfy that for every $i \in \mathbb{I}$,

- $$D_{\log T_i, i} = \begin{cases} g_1^{\sum_{j \in [1, d_\iota - 1]} (A_j + A'_j) i^j + \alpha} g^{\delta_i + \delta'_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s_{ji} + \tilde{s}_{ji}} & \\ \qquad \text{(if } k_i = \log T_i\text{)}, & \\ g_1^{\sum_{j \in [1, d_\iota - 1]} (A_j + A'_j) i^j + \alpha} g^{\delta_i + \delta'_i} \prod_{j \in [0, \log T_i - 2]} (u_{ji} v_0^{R_i[j]})^{s_{ji} + \tilde{s}_{ji}} & \\ \cdot (u_{\log T_i - 1, i} v_0)^{s'_{\log T_i, i} + \tilde{s}'_{\log T_i, i}} & \\ \qquad \text{(if } k_i = \log T_i - 1\text{)}, & \\ g_1^{\sum_{j \in [1, d_\iota - 1]} (A_j + A'_j) i^j + \alpha} g^{\delta_i + \delta'_i} \prod_{j \in [0, k_i - 1]} (u_{ji} v_0^{R_i[j]})^{s_{ji} + \tilde{s}_{ji}} & \\ \cdot (u_{k_i, i} v_0)^{s'_{k_i, i} + \tilde{s}'_{k_i, i}} \prod_{j \in [k_i + 1, \log T_i - 1]} (u_{ji} v_0^{R_i[j]})^{s^\dagger_{ji}} & \\ \qquad \text{(otherwise)}, & \end{cases}$$
- $$E_{\log T_i, i} = \begin{cases} g^{-\delta_i - \delta'_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji} + \tilde{t}_{ji}} & \text{(if } k'_i = \log T_i\text{)}, \\ g^{-\delta_i - \delta'_i} \prod_{j \in [0, \log T_i - 2]} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji} + \tilde{t}_{ji}} \cdot (w_{\log T_i - 1, i} v_0)^{t'_{\log T_i, i} + \tilde{t}'_{\log T_i, i}} & \\ \qquad \text{(if } k'_i = \log T_i - 1\text{)}, & \\ g^{-\delta_i - \delta'_i} \prod_{j \in [0, k'_i - 1]} (w_{ji} v_0^{\hat{L}_i[j]})^{t_{ji} + \tilde{t}_{ji}} \cdot (w_{k'_i, i} v_0)^{t'_{k'_i, i} + \tilde{t}'_{k'_i, i}} & \\ \cdot \prod_{j \in [k'_i + 1, \log T_i - 1]} (w_{ji} v_0^{\hat{L}_i[j]})^{t^\dagger_{ji}} & \\ \qquad \text{(otherwise)}, & \end{cases}$$

and for every $i \in \mathbb{I}$ and every $j \in [0, \log T_i - 1]$,

$$d_{ji} = \begin{cases} g^{s_{ji} + \tilde{s}_{ji}} & \text{(if } j \in [0, k_i - 1]\text{)} \\ g^{s'_{ji} + \tilde{s}'_{ji}} & \text{(if } j = k_i\text{)} \\ g^{s^\dagger_{ji}} & \text{(otherwise)} \end{cases} \quad \text{and } e_{ji} = \begin{cases} g^{t_{ji} + \tilde{t}_{ji}} & \text{(if } j \in [0, k'_i - 1]\text{)} \\ g^{t'_{ji} + \tilde{t}'_{ji}} & \text{(if } j = k'_i\text{)} \\ g^{t^\dagger_{ji}} & \text{(otherwise)} \end{cases},$$

where any randomness from

- $\{A'_j \mid j \in [1, d_\iota - 1]\}$, $\{\delta'_i \mid i \in \mathbb{I}\}$
- $\{\tilde{s}_{ji} \mid i \in \mathbb{I}, j \in [0, k_i - 1]\}$
- $\{\tilde{s}'_{k_i, i} \mid i \in \mathbb{I} \text{ s.t. } k_i \in [0, \log T_i - 1]\}$
- $\{s^\dagger_{ji} \mid i \in \mathbb{I} \text{ s.t. } k_i \in [0, \log T_i - 2], j \in [k_i + 1, \log T_i - 1]\}$
- $\{\tilde{t}_{ji} \mid i \in \mathbb{I}, j \in [0, k'_i - 1]\}$
- $\{\tilde{t}'_{k'_i, i} \mid i \in \mathbb{I} \text{ s.t. } k'_i \in [0, \log T_i - 1]\}$
- $\{t^\dagger_{ji} \mid i \in \mathbb{I} \text{ s.t. } k'_i \in [0, \log T_i - 2], j \in [k'_i + 1, \log T_i - 1]\}$

is chosen uniformly at random from $\mathbb{Z}_p$.

Note that for every $i \in \mathbb{I}$, there exist $\log T_i$ elements $\{H_{ji} \mid i \in \mathbb{I}, j \in [0, \log T_i - 1]\}$ from $\mathbb{G}$ which satisfy $D_{\log T_i, i} = g_1^{\sum_{j\in[1,d_\iota-1]}(A_j+A'_j)\cdot i^j+\alpha} g^{\delta_i+\delta'_i} \prod_{j\in[0,\log T_i-1]} H_{ji}$. Obviously, H_{ji} distributes identically to $(u_{ji}v_0^{R_i[j]})^h$ for $h \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$. Moreover, for every $i \in \mathbb{I}$, there exist $\log T_i$ elements $\{H'_{ji} \mid i \in \mathbb{I}, j \in [0, \log T_i - 1]\}$ from $\mathbb{G}$ which satisfy $E_{\log T_i, i} = g^{-\delta_i-\delta'_i} \prod_{j\in[0,\log T_i-1]} H'_{ji}$. Obviously, H_{ji} distributes identically to $(w_{ji}v_0^{\hat{L}_i[j]})^h$ for $h \overset{\mathrm{U}}{\longleftarrow} \mathbb{Z}_p$.

Using $\{sk_{l'_i}, sk_{r'_i} \mid i \in \mathbb{I}\}$, we generate a signature σ. It consists of $(U, \{V_{ji}, V'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, W)$, where

$$\begin{aligned}
U &:= g_1^{\alpha} \prod_{i\in[1,D]\setminus\mathbb{I}} \prod_{j\in[0,\log T_i-1]} (u_{ji}v_0^{R_i[j]})^{s^*_{ji}} (w_{ji}v_0^{\hat{L}_i[j]})^{t^*_{ji}} \\
&\quad \cdot \prod_{i\in\mathbb{I}} \prod_{j\in[0,\log T_i-1]} H_{ji}H'_{ji} \cdot \left(u \prod_{j\in[0,N-1]} v_j^{m[j]}\right)^r, \\
(V_{ji}, V'_{ji}) &:= \begin{cases} (d_{ji}, e_{ji}) & \text{(if } i \in \mathbb{I}\text{)}, \\ (g^{s^*_{ji}}, g^{t^*_{ji}}) & \text{(otherwise)}, \end{cases} \\
W &:= g^r.
\end{aligned}$$

Note that any randomness from r and $\{s^*_{ji}, t^*_{ji} \mid i \in [1, D] \setminus \mathbb{I}, j \in [0, \log T_i - 1]\}$ is chosen uniformly at random from $\mathbb{Z}_p$. Hence, the signature distributes identically to one in $\boldsymbol{Expt}^{\mathsf{PP}}_{\Pi_{\mathrm{SB}},\mathcal{A},1}$. □

4 Multi-Dimensional *Super*-Range Signatures (MDSPRS)

In (one-dimensional) super-range signatures, the key-range of a signer must be a super-range of a signature-range. We remind us that, in (one-dimensional) sub-range signatures in Sect. 3, the former must be a sub-range of the latter. In this section, we formally define syntax and security of multi-dimensional super-range signatures (MDSPRS), and propose a secure construction.

Syntax. Multi-Dimensional *Super*-Range Signatures (MDSPRS) consist of the following 5 polynomial time algorithms, where `Ver` is deterministic and the others are probabilistic.

Setup `Setup`: The setup algorithm is defined analogously to the one of MDS-BRS. See Sect 3.

Key-generation `KGen`: It takes msk, key-ranges $\{L_i, R_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i\in[1,D]} 0 \leq L_i \leq R_i \leq T_i - 1$ and a threshold $d \in [1, D]$, then outputs a secret-key sk for the key-ranges. Concisely, $sk \leftarrow \mathtt{KGen}(msk, \{L_i, R_i \mid i \in [1, D], d)$.

Key-delegation $\texttt{KDel}$**:** It takes sk for $\{L_i, R_i \mid i \in [1, D]\}$ and $d \in [1, D]$, shrunk key-ranges $\{L'_i, R'_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i\in[1,D]} 0 \le L_i \le L'_i \le R'_i \le R_i \le T_i - 1$, then outputs a secret-key sk' for the shrunk ones. We write $sk' \leftarrow \texttt{KDel}(sk, \{L'_i, R'_i \mid i \in [1, D]\})$.

Signing $\texttt{Sig}$**:** It takes sk for $\{L_i, R_i \mid i \in [1, D]\}$ and $d \in [1, D]$, a message $m \in \{0, 1\}^*$, and signature-ranges $\{l_i, r_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i\in[1,D]} 0 \le l_i \le r_i \le T_i - 1$, then outputs a signature σ. We write $\sigma \leftarrow \texttt{Sig}(sk, m, \{l_i, r_i \mid i \in [1, D]\})$.

Verification $\texttt{Ver}$**:** It takes σ, $m \in \{0, 1\}^*$ and $\{l_i, r_i \mid i \in [1, D]\}$, then outputs a bit 1/0. We write $1/0 \leftarrow \texttt{Ver}(\sigma, m, \{l_i, r_i \mid i \in [1, D]\})$.

We require every MDSPRS scheme to be correct. An MDSPRS scheme $\Sigma_{\mathrm{SP}} = \{\texttt{Setup}, \texttt{KGen}, \texttt{KDel}, \texttt{Sig}, \texttt{Ver}\}$ is correct, if $\forall\lambda \in \mathbb{N}$, $\forall D \in \mathbb{N}$, $\forall T_1 \in \mathbb{N}, \cdots, \forall T_D \in \mathbb{N}$, $\forall(mpk, msk) \leftarrow \texttt{Setup}(1^\lambda, D, \{T_i \mid i \in \mathbb{N}\})$, $\forall\{L_i, L'_i, R_i, R'_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i\in[1,D]} 0 \le L_i \le L'_i \le R'_i \le R_i \le T_i - 1$, $\forall d \in [1, D]$, $\forall sk \leftarrow \texttt{KGen}(msk, \{L_i, R_i \mid i \in [1, D]\}, d)$, $\forall sk' \leftarrow \texttt{KDel}(sk, \{L'_i, R'_i \mid i \in [1, D]\})$, $\forall m \in \{0, 1\}^*$, $\forall\{l_i, r_i \mid i \in [1, D]\}$ s.t. $\bigwedge_{i\in[1,D]} 0 \le l_i \le r_i \le T_i - 1 \bigwedge \sum_{i\in[1,D] \text{ s.t. } L'_i \le l_i \le r_i \le R'_i} 1 \ge d$, $\forall\sigma \leftarrow \texttt{Sig}(sk', m, \{l_i, r_i \mid i \in [1, D]\})$, $1 \leftarrow \texttt{Ver}(\sigma, m, \{l_i, r_i \mid i \in [1, D]\})$.

Existential Unforgeability and Perfect Privacy. Analogously to MDSBRS, for an MDSPRS scheme Σ_{SP} and a probabilistic algorithm $\mathcal{A}$, we consider experiments for existential unforgeability and perfect privacy depicted in Fig. 5.

Definition 4. *An MDSPRS scheme* Σ_{SP} *is (adaptively) existentially unforgeable, if* $\forall\lambda \in \mathbb{N}$, $\forall D \in \mathbb{N}$, $\forall T_1 \in \mathbb{N}, \cdots, \forall T_D \in \mathbb{N}$, $\forall\mathcal{A} \in \mathbb{PPT}_\lambda$, $\exists\epsilon \in \mathbb{NGL}_\lambda$ *s.t.* $\boldsymbol{Adv}^{\textit{EUF-CMA}}_{\Sigma_{\mathrm{SP}},\mathcal{A},D,\{T_i \mid i\in[1,D]\}}(\lambda) := \Pr[1 \leftarrow \boldsymbol{Expt}^{\textit{EUF-CMA}}_{\Sigma_{\mathrm{SP}},\mathcal{A}}(1^\lambda, D, \{T_i \mid i \in [1, D]\})] < \epsilon$.

Definition 5. *An MDSPRS scheme* Σ_{SP} *is perfectly private, if for every* $\lambda \in \mathbb{N}$, *every* $D \in \mathbb{N}$, *every* $T_1 \in \mathbb{N}, \cdots,$ *every* $T_D \in \mathbb{N}$ *and every probabilistic algorithm* $\mathcal{A}$, *there exist probabilistic polynomial time algorithms* $\{\widehat{\texttt{Setup}}, \widehat{\texttt{KGen}}, \widehat{\texttt{KDel}}, \widehat{\texttt{Sig}}\}$ *such that* $\boldsymbol{Adv}^{\textit{PP}}_{\Sigma_{\mathrm{SP}},\mathcal{A},D,\{T_i \mid i\in[1,D]\}}(\lambda) := |\Pr[1 \leftarrow \boldsymbol{Expt}^{\textit{PP}}_{\Sigma_{\mathrm{SP}},\mathcal{A},0}(1^\lambda, D, \{T_i \mid i \in [1, D]\})] - \Pr[1 \leftarrow \boldsymbol{Expt}^{\textit{PP}}_{\Sigma_{\mathrm{SP}},\mathcal{A},1}(1^\lambda, D, \{T_i \mid i \in [1, D]\})]| = 0$.

4.1 Our MDSPRS Scheme

The technique of our MDSPRS scheme is basically the same as the one of our MDSBRS scheme. A secret-key for a D-dimensional key-range $\{[L_i, R_i] \mid i \in [1, D]\}$ consists of $2D$ number of partial secret-keys. The partial secret-key for L_i (resp. R_i) is a randomly-generated CS-FSS (resp. CS-BSS) secret-key for L_i (resp. R_i) based on one (resp. the other one) of the two second-level pseudo master secret-keys. Our MDSPRS scheme $\Pi_{\mathrm{SP}} = \{\texttt{Setup}, \texttt{KGen}, \texttt{KDel}, \texttt{Sig}, \texttt{Ver}\}$ is formally described in Sect. B.

```
$\boldsymbol{Expt}^{\texttt{EUF-CMA}}_{\Sigma_{\mathrm{SP}},\mathcal{A}}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$:
  $(mpk, msk) \leftarrow \texttt{Setup}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$
  $(\sigma^*, \{l_i^*, r_i^* \mid i \in [1,D]\}, m^*) \leftarrow \mathcal{A}^{\mathfrak{Reveal},\mathfrak{Sign}}(mpk)$, where
  ------------------------------------------------------------
  $-\mathfrak{Reveal}(\{L_{i,\iota}, R_{i,\iota} \mid i \in [1,D]\}, d_\iota)$: // $\iota \in [1, q_r]$
    Rtrn $sk_\iota \leftarrow \texttt{KGen}(msk, \{L_{i,\iota}, R_{i,\iota} \mid i \in [1,D]\}, d_\iota)$.
  $-\mathfrak{Sign}(\{L_{i,\theta}, R_{i,\theta} \mid i \in [1,D]\}, d_\theta, \{l_{i,\theta}, r_{i,\theta} \mid i \in [1,D]\}, m_\theta \in \{0,1\}^*)$: // $\theta \in [1, q_s]$
    $sk_\theta \leftarrow \texttt{KGen}(msk, \{L_{i,\theta}, R_{i,\theta} \mid i \in [1,D]\}, d_\theta)$.
    Rtrn $\sigma_\theta \leftarrow \texttt{Sig}(sk_\theta, \{l_{i,\theta}, r_{i,\theta} \mid i \in [1,D]\}, m_\theta)$.
  ------------------------------------------------------------
  Rtrn 1 if $1 \leftarrow \texttt{Ver}(\sigma^*, \{l_i^*, r_i^* \mid i \in [1,D]\}, m^*)$
    $\bigwedge_{\iota \in [1,q_r]} \left( \sum_{i \in [1,D] \text{ s.t. } L_{i,\iota} \leq l_i^* \leq r_i^* \leq R_{i,\iota}} 1 \right) < d_\iota$
    $\bigwedge_{\theta \in [1,q_s]} (\{l_{i,\theta}, r_{i,\theta} \mid i \in [1,D]\}, m_\theta) \neq (\{l_i^*, r_i^* \mid i \in [1,D]\}, m^*)$.
  Rtrn 0.
$\boldsymbol{Expt}^{\texttt{PP}}_{\Sigma_{\mathrm{SP}},\mathcal{A},\beta}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$:   // $\beta \in \{0, 1\}$
  $(mpk, msk) \leftarrow \texttt{Setup}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$.
  $(mpk, \widehat{msk}) \leftarrow \widehat{\texttt{Setup}}(1^\lambda, D, \{T_i \mid i \in [1,D]\})$.
  Rtrn $b \leftarrow \mathcal{A}^{\mathfrak{Reveal},\mathfrak{Sign}}(mpk, msk)$, where
  ------------------------------------------------------------
  $-\mathfrak{Reveal}(\{L_{i,\iota}, R_{i,\iota} \mid i \in [1,D]\}, d_\iota)$: // $\iota \in [1, q_r]$
    Rtrn $\perp$ if $\neg \left[ \bigwedge_{i \in [1,D]} 0 \leq L_{i,\iota} \leq R_{i,\iota} \leq T_i - 1 \right] \bigvee d_\iota \notin [1,D]$.
    $sk_\iota \leftarrow \texttt{KGen}(msk, \{L_{i,\iota}, R_{i,\iota} \mid i \in [1,D]\}, d_\iota)$.
    $sk_\iota \leftarrow \widehat{\texttt{KGen}}(\widehat{msk}, \{L_{i,\iota}, R_{i,\iota} \mid i \in [1,D]\}, d_\iota)$. Rtrn $sk_\iota$.
  $-\mathfrak{Delegate}(\iota \in [1, q_r], \{L_i', R_i' \mid i \in [1,D]\})$:
    Rtrn $\perp$ if $\neg \left[ \bigwedge_{i \in [1,D]} 0 \leq L_{i,\iota} \leq L_i' \leq R_i' \leq R_{i,\iota} \leq T_i - 1 \right]$.
    For every $i \in [1,D]$, $(L_{i,\iota}, R_{i,\iota}) := (L_i', R_i')$.
    $sk_\iota' \leftarrow \texttt{KDel}(sk_\iota, \{L_i', R_i' \mid i \in [1,D]\})$.
    $sk_\iota' \leftarrow \widehat{\texttt{KDel}}(sk_\iota, \{L_i', R_i' \mid i \in [1,D]\})$. Rtrn $sk_\iota := sk_\iota'$.
  $-\mathfrak{Sign}(\iota \in [1, q_r], m, \{l_i, r_i \mid i \in [1,D]\})$:
    Rtrn $\perp$ if $\neg \left[ \bigwedge_{i \in [1,D]} 0 \leq l_i \leq r_i \leq T_i - 1 \right] \bigvee \left( \sum_{i \in [1,D] \text{ s.t. } L_{i,\iota} \leq l_i \leq r_i \leq R_{i,\iota}} 1 \right) < d_\iota$.
    $\sigma \leftarrow \texttt{Sig}(sk_\iota, \{l_i, r_i \mid i \in [1,D]\}, m)$.
    $\sigma \leftarrow \widehat{\texttt{Sig}}(\widehat{msk}, \{l_i, r_i \mid i \in [1,D]\}, m)$. Rtrn $\sigma$.
```

Fig. 5. Security experiments w.r.t. an MDSPRS scheme Σ_{SP}. Top: (Adaptive) existential unforgeability. Bottom: Perfect privacy.

4.2 Existential Unforgeability and Perfect Privacy of Our MDSPRS Scheme

Security of our MDSPRS scheme Π_{SP} is guaranteed by the following 2 theorems. Their proofs are omitted since they are analogous to the ones for our MDSBRS scheme Π_{SB}.

Theorem 3. *Π_{SP} is existentially unforgeable under the co-CDH assumption.*

Theorem 4. *Π_{SP} is perfectly private.*

5 Conclusion

We proposed multi-dimensional sub-range signatures (MDSBRS) as a generalization of time-specific signatures (TSS) [10,5]. MDSBRS are a generalization of TSS because of the following properties, namely (1) Each secret-key is associated with a range, (2) Key-delegatability, and (3) Multi-dimensionalization with threshold signability. As a related primitive, we also proposed multi-dimensional super-range signatures (MDSPRS).

Our concrete MDSBRS scheme is a generalization of the TSS scheme in [5] based on forward-secure signatures. We also proposed a concrete MDSPRS scheme based on the same technique. Efficiency of the schemes is analysed as shown in Table 1. To the best of our knowledge, our schemes are the first currently-known poly-logarithmically efficient ones.

Table 1. Efficiency of our MDSBRS and MDSPRS schemes.

Schemes	$\lvert mpk\rvert$	$\lvert msk\rvert$	$\lvert sk\rvert$	$\lvert\sigma\rvert$
Π_{SB}	$\left(3+2\sum_{i=1}^{D}\log T_i+N\right)(\lvert g\rvert+\lvert\tilde{g}\rvert)$	$\lvert g\rvert$	$\mathcal{O}\left(\sum_{i=1}^{D}\log T_i\right)\lvert g\rvert$	$\left(2+2\sum_{i=1}^{D}\log T_i\right)\lvert g\rvert$
Π_{SP}				

For a data a, $|a|$ denotes its bit length. $|g|$ (resp. $|\tilde{g}|$) denotes bit length of an element in bilinear group $\mathbb{G}$ (resp. $\tilde{\mathbb{G}}$).

References

1. J. Blömer, F. Eidens, and J. Juhnke. Enhanced security of attribute-based signatures. In *CANS 2018*, volume 11124 of LNCS, pages 235–255. Springer, 2018.
2. R. Canetti, S. Halevi, and J. Katz. A forward-secure public-key encryption scheme. In *EUROCRYPT 2003*, volume 2656 of LNCS, pages 255–271. Springer, 2003.
3. S. Chatterjee and P. Sarkar. Practical hybrid (hierarchical) identity-based encryption schemes based on the decisional bilinear diffie-hellman assumption. *International Journal of Applied Cryptography (IJACT)*, 3(1):47–83, 2013.

4. M. Ishizaka and S. Kiyomoto. Time-specific encryption with constant size secret-keys secure under standard assumption. Cryptology ePrint Archive: Report 2020/595, 2020.
5. M. Ishizaka and S. Kiyomoto. Time-specific signatures. In *ISC 2020*, volume 12472 of LNCS, pages 20–38. Springer, 2020.
6. K. Kasamatsu, T. Matsuda, K. Emura, N. Attrapadung, G. Hanaoka, and H. Imai. Time-specific encryption from forward-secure encryption. In *SCN 2012*.
7. K. Kasamatsu, T. Matsuda, K. Emura, N. Attrapadung, G. Hanaoka, and H. Imai. Time-specific encryption from forward-secure encryption: generic and direct constructions. *International Journal of Information Security*, 15(5):549–571, 2016.
8. K. Kasamatsu, T. Matsuda, G. Hanaoka, and H. Imai. Ciphertext policy multi-dimensional range encryption. In *ICISC 2012*, volume 7839 of LNCS, pages 247–261. Springer, 2012.
9. H.K. Maji, M. Prabhakaran, and M. Rosulek. Attribute-based signatures. In *CT-RSA 2011*, volume 6558 of LNCS, pages 376–392. Springer, 2011.
10. K. G. Paterson and E. A. Quaglia. Time-specific encryption. In *SCN 2010*, volume 6280 of LNCS, pages 1–16. Springer, 2010.
11. A. Sahai and B. Waters. Fuzzy identity-based encryption. In *EUROCRYPT 2005*, volume 3494 of LNCS, pages 457–473. Springer, 2005.
12. Y. Sakai, N. Attrapadung, and G. Hanaoka. Attribute-based signatures for circuits from bilinear map. In *PKC 2016*, volume 9612 of LNCS, pages 283–300. Springer, 2016.
13. A. Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

A A TSS Scheme by Ishizaka and Kiyomoto [5]

An FSS-based TSS scheme $\mathit{\Pi}_{\mathrm{TSS}} = \{\mathtt{Setup}, \mathtt{KGen}, \mathtt{Sig}, \mathtt{Ver}\}$ proposed in [5] is formally described as follows.

$\mathtt{Setup}\left(1^\lambda, N, T\right)$**:**

- $(p, \mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T, e, g, \tilde{g}) \leftarrow \mathcal{G}_{BG}(1^\lambda)$. $\alpha \xleftarrow{\mathrm{U}} \mathbb{Z}_p$, $g_2 := \tilde{g}^\alpha$. $g_1 \xleftarrow{\mathrm{U}} \mathbb{G}$.
- For $i \in [0, \log T - 1]$: $x_i, z_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$, $u_i := g^{x_i}, \tilde{u}_i := \tilde{g}^{x_i}, w_i := g^{z_i}, \tilde{w}_i := \tilde{g}^{z_i}$.
- $x_{\log T} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$, $u_{\log T} := g^{x_{\log T}}$, $\tilde{u}_{\log T} := \tilde{g}^{x_{\log T}}$.
- For $i \in [0, N-1]$: $y_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$, $v_i := g^{y_i}, \tilde{v}_i := \tilde{g}^{y_i}$.
- $mpk := \begin{pmatrix} p, \mathbb{G}, \tilde{\mathbb{G}}, \mathbb{G}_T, e, g, \tilde{g}, g_1, g_2, \{u_i, \tilde{u}_i, w_i, \tilde{w}_i \mid i \in [0, \log T - 1]\}, \\ u_{\log T}, \tilde{u}_{\log T}, \{v_i, \tilde{v}_i \mid i \in [0, N-1]\} \end{pmatrix}$.
- $msk := g_1^\alpha$.
- **Rtrn** (mpk, msk).

$\mathtt{KGen}\left(msk, t\right)$**:**

- **Rtrn** $\perp$ if $t \notin [0, T-1]$.
- $\delta \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. $\tilde{t} := T - 1 - t$.

- For $i \in [0, \log T - 1]$: $r_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. If $t[i] = 0$, $r'_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $sk_r := \left(\begin{array}{c} g_1^{\alpha} g^{\delta} \prod\limits_{i \in [0, \log T - 1]} \left(u_i v_0^{t[i]} \right)^{r_i}, g^{r_0}, \cdots, g^{r_{\log T - 1}}, \\ \left\{ g_1^{\alpha} g^{\delta} \prod\limits_{i \in [0, j-1]} \left(u_i v_0^{t[i]} \right)^{r_i} (u_j v_0)^{r'_j}, g^{r'_j} \middle| \begin{array}{c} j \in [0, \log T - 1] \\ \text{s.t. } t[j] = 0 \end{array} \right\} \end{array} \right)$.
- For $i \in [0, \log T - 1]$: $s_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. If $\tilde{t}[i] = 0$, $s'_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $sk_l := \left(\begin{array}{c} g^{-\delta} \prod\limits_{i \in [0, \log T - 1]} \left(w_i v_0^{\tilde{t}[i]} \right)^{s_i}, g^{s_0}, \cdots, g^{s_{\log T - 1}}, \\ \left\{ g^{-\delta} \prod\limits_{i \in [0, j-1]} \left(w_i v_0^{\tilde{t}[i]} \right)^{s_i} (w_j v_0)^{s'_j}, g^{s'_j} \middle| \begin{array}{c} j \in [0, \log T - 1] \\ \text{s.t. } \tilde{t}[j] = 0 \end{array} \right\} \end{array} \right)$.
- **Rtrn** $sk_t := (sk_l, sk_r)$.

`Sig` (sk_t, m, L, R)**:**

- **Rtrn** $\perp$ if $\neg[m \in \{0,1\}^N \wedge 0 \leq L \leq R \leq T - 1]$.
- Parse sk_t as (sk_l, sk_r). $\tilde{t} := T - 1 - t$. $\tilde{L} := T - 1 - L$.
- Parse sk_r as $\left(D_{\log T}, d_0, \cdots, d_{\log T - 1}, \left\{D_j, d'_j \mid j \in [0, \log T - 1] \text{ s.t. } t[j] = 0\right\}\right)$.
- Parse sk_l as $\left(E_{\log T}, e_0, \cdots, e_{\log T - 1}, \left\{E_j, e'_j \mid j \in [0, \log T - 1] \text{ s.t. } \tilde{t}[j] = 0\right\}\right)$.
- $t \in [L, R]$ implies

$$\begin{array}{c} \left[\begin{array}{c} \exists i_r \in [0, \log T] \text{ s.t.} \\ \bigwedge\limits_{i \in [0, i_r - 1]} [t[i] = R[i]] \bigwedge \left[i_r \neq \log T \implies t[i_r] = 0 \bigwedge R[i_r] = 1 \right] \end{array} \right] \\ \bigwedge \left[\begin{array}{c} \exists i_l \in [0, \log T] \text{ s.t.} \\ \bigwedge\limits_{i \in [0, i_l - 1]} \left[\tilde{t}[i] = \tilde{L}[i] \right] \bigwedge \left[i_l \neq \log T \implies \tilde{t}[i_l] = 0 \bigwedge \tilde{L}[i_l] = 1 \right] \end{array} \right]. \end{array}$$

- For $i \in [0, i_r]$, $\tilde{r}_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. For $i \in [i_r + 1, \log T - 1]$, $r^*_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- For $i \in [0, i_l]$, $\tilde{s}_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. For $i \in [i_l + 1, \log T - 1]$, $s^*_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $r_{\log T} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- **Rtrn** $\sigma :=$

$$\left(\begin{array}{c} D_{i_r} \prod\limits_{i \in [0, i_r]} \left(u_i v_0^{R[i]} \right)^{\tilde{r}_i} \prod\limits_{i \in [i_r + 1, \log T - 1]} \left(u_i v_0^{R[i]} \right)^{r^*_i} \\ \cdot E_{i_l} \prod\limits_{i \in [0, i_l]} \left(w_i v_0^{\tilde{L}[i]} \right)^{\tilde{s}_i} \prod\limits_{i \in [i_l + 1, \log T - 1]} \left(w_i v_0^{\tilde{L}[i]} \right)^{s^*_i} \left(u_{\log T} \prod\limits_{j \in [0, N-1]} v_j^{m[j]} \right)^{r_{\log T}}, \\ \left\{ d_i g^{\tilde{r}_i} \mid i \in [0, i_r - 1] \right\}, d'_{i_r} g^{\tilde{r}_{i_r}}, \left\{ g^{r^*_i} \mid i \in [i_r + 1, \log T - 1] \right\}, \\ \left\{ e_i g^{\tilde{s}_i} \mid i \in [0, i_l - 1] \right\}, e'_{i_l} g^{\tilde{s}_{i_l}}, \left\{ g^{s^*_i} \mid i \in [i_l + 1, \log T - 1] \right\}, g^{r_{\log T}} \end{array} \right).$$

`Ver` (σ, m, L, R)**:**

- **Rtrn** $\perp$ if $\neg[m \in \{0,1\}^N \wedge 0 \leq L \leq R \leq T-1]$.
- Parse σ as $\left(U, V_0, \cdots, V_{\log T-1}, V'_0, \cdots, V'_{\log T-1}, V_{\log T}\right)$.
- $\tilde{L} := T - 1 - L$.
- **Rtrn** 1 if

$$(U, \tilde{g}) = e\left(g_1, g_2\right) \prod_{i \in [0, \log T-1]} e\left(V_i, \tilde{u}_i \tilde{v}_0^{R[i]}\right) e\left(V'_i, \tilde{w}_i \tilde{v}_0^{\tilde{L}[i]}\right)$$
$$e\left(V_{\log T}, \tilde{u}_{\log T} \prod_{j \in [0, N-1]} \tilde{v}_j^{m[j]}\right).$$

- **Rtrn** 0, otherwise.

B Our MDSPRS Scheme

The scheme $\mathit{\Pi}_{\mathrm{SP}} = \{$`Setup`, `KGen`, `KDel`, `Sig`, `Ver`$\}$ is formally described as follows.

`Setup`$(1^\lambda, D, T_1, \cdots, T_D)$**:** The same as the one of our MDSBRS scheme. See Subsect. 3.2.

`KGen`$(msk, L_1, R_1, \cdots, L_D, R_D, d)$**:** It returns $\perp$ if $\neg[1 \leq d \leq D \wedge_{i=0}^{D} 0 \leq L_i \leq R_i \leq T_i - 1]$.
It chooses $A_1, \cdots, A_{d-1}$ uniformly at random from $\mathbb{Z}_p$. A $(d-1)$-dimensional polynomial $f : [1, D] \to \mathbb{Z}_p$ is defined as $f(x) := \sum_{j \in [1, d-1]} A_j x^j + \alpha$. For every $i \in [1, D]$, it does:

- $\delta_i \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. $\hat{R}_i := T_i - 1 - R_i$.
- For $j \in [0, \log T_i - 1]$: $s_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. If $r_i[j] = 0$, $s'_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $sk_{L_i} := \left(\begin{array}{c} g_1^{f(i)} g^{\delta_i} \prod\limits_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{L_i[j]})^{s_{ji}}, g^{s_{0,i}}, \cdots, g^{s_{\log T_i - 1, i}}, \\ \left\{ g_1^{f(i)} g^{\delta_i} \prod\limits_{j \in [0, k-1]} (u_{ji} v_0^{L_i[j]})^{s_{ji}} (u_{ki} v_0)^{s'_{ki}}, g^{s'_{ki}} \,\middle|\, \begin{array}{c} k \in [0, \log T_i - 1] \\ \text{s.t. } L_i[k] = 0 \end{array} \right\} \end{array}\right)$.
- For $j \in [0, \log T_i - 1]$: $t_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. If $\hat{l}_i[j] = 0$, $t'_{ji} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.
- $sk_{R_i} := \left(\begin{array}{c} g^{-\delta_i} \prod\limits_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{R}_i[j]})^{t_{ji}}, g^{t_{0,i}}, \cdots, g^{t_{\log T_i - 1, i}}, \\ \left\{ g^{-\delta_i} \prod\limits_{j \in [0, k-1]} (w_{ji} v_0^{\hat{R}_i[j]})^{t_{ji}} (w_{ki} v_0)^{t'_{ki}}, g^{t'_{ki}} \,\middle|\, \begin{array}{c} k \in [0, \log T_i - 1] \\ \text{s.t. } \hat{R}_i[k] = 0 \end{array} \right\} \end{array}\right)$.

Then, it returns $sk_: = (\{sk_{L_i}, sk_{R_i} \mid i \in [1, D]\}, d)$.

$\texttt{KUpd}_{\beta}(sk_r, r, R, i)$**:** // $\beta \in \{0, 1\}$

The same as the one of our MDSBRS scheme. See Subsect. 3.2.

$\texttt{KDel}(sk, L'_1, R'_1, \cdots, L'_D, R'_D)$**:** It parses sk for $\{L_i, R_i \mid i \in [1, D]\}$ and d as $(\{sk_{L_i}, sk_{R_i} \mid i \in [1, D]\}, d)$. It returns $\perp$ if $\neg[\bigwedge_{i=1}^{D} 0 \leq L_i \leq L'_i \leq R'_i \leq R_i \leq T_i - 1]$.

Firstly, it re-randomizes sk. Let $A'_1, \cdots, A'_{d-1} \overset{\text{U}}{\longleftarrow} \mathbb{Z}_p$. For every $i \in [1, D]$, it does:

- $\delta'_i \overset{\text{U}}{\longleftarrow} \mathbb{Z}_p$. $\hat{l}_i := T_i - 1 - l_i$.
- Parse sk_{L_i} as $(D_{\log T_i, i}, d_{0,i}, \cdots, d_{\log T_i - 1, i}, \{D_{k,i}, d'_{k,i} \mid k \in [0, \log T_i - 1]$ s.t. $L_i[k] = 0\})$.
- For $j \in [0, \log T_i - 1]$: $\tilde{s}_{ji} \overset{\text{U}}{\longleftarrow} \mathbb{Z}_p$. If $L_i[j] = 0$, $\tilde{s}'_{ji} \overset{\text{U}}{\longleftarrow} \mathbb{Z}_p$.
- Let $\tilde{sk}_{r_i} := (\tilde{D}_{\log T_i, i}, \tilde{d}_{0,i}, \cdots, \tilde{d}_{\log T_i - 1, i}, \{\tilde{D}_{k,i}, \tilde{d}'_{k,i} \mid k \in [0, \log T_i - 1]$ s.t. $L_i[k] = 0\})$, where
  - $\tilde{D}_{\log T_i, i} := D_{\log T_i, i} \cdot g_1^{\sum_{j \in [1, d-1]} A'_j \cdot i^j} g^{\delta'_i} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{L_i[j]})^{\tilde{s}_{ji}}$.
  - $\tilde{d}_{j,i} := d_{j,i} \cdot g^{\tilde{s}_{j,i}}$ (for each $j \in [0, \log T_i - 1]$).
  - $\tilde{D}_{k,i} := D_{k,i} \cdot g_1^{\sum_{j \in [1, d-1]} A'_j \cdot i^j} g^{\delta'_i} \prod_{j \in [0, k-1]} (u_{ji} v_0^{L_i[j]})^{\tilde{s}_{ji}} (u_{ki} v_0)^{\tilde{s}'_{ki}}$ and $\tilde{d}'_{k,i} := d'_{k,i} \cdot g^{\tilde{s}'_{ki}}$ (for each $k \in [0, \log T_i - 1]$ s.t. $L_i[k] = 0$).
- Parse sk_{R_i} as $(E_{\log T_i, i}, e_{0,i}, \cdots, e_{\log T_i - 1, i}, \{E_{k,i}, e'_{k,i} \mid k \in [0, \log T_i - 1]$ s.t. $\hat{R}_i[k] = 0\})$.
- For $j \in [0, \log T_i - 1]$: $\tilde{t}_{ji} \overset{\text{U}}{\longleftarrow} \mathbb{Z}_p$. If $\hat{R}_i[j] = 0$, $\tilde{t}'_{ji} \overset{\text{U}}{\longleftarrow} \mathbb{Z}_p$.
- Let $\tilde{sk}_{l_i} := (\tilde{E}_{\log T_i, i}, \tilde{e}_{0,i}, \cdots, \tilde{e}_{\log T_i - 1, i}, \{\tilde{E}_{k,i}, \tilde{e}'_{k,i} \mid k \in [0, \log T_i - 1]$ s.t. $\hat{R}_i[k] = 0\})$, where
  - $\tilde{E}_{\log T_i, i} := E_{\log T_i, i} \cdot g^{-\delta'_i} \prod_{j \in [0, \log T_i - 1]} (w_{ji} v_0^{\hat{R}_i[j]})^{\tilde{t}_{ji}}$.
  - $\tilde{e}_{j,i} := e_{j,i} \cdot g^{\tilde{t}_{j,i}}$ (for each $j \in [0, \log T_i - 1]$).
  - $\tilde{E}_{k,i} := E_{k,i} \cdot g^{-\delta'_i} \prod_{j \in [0, k-1]} (w_{ji} v_0^{\hat{R}_i[j]})^{\tilde{t}_{ji}} (w_{ki} v_0)^{\tilde{t}'_{ki}}$ and $\tilde{e}'_{k,i} := e'_{k,i} \cdot g^{\tilde{t}'_{ki}}$ (for each $k \in [0, \log T_i - 1]$ s.t. $\hat{R}_i[k] = 0$).

Secondly, it updates the re-randomized secret-key for $\{L_i, R_i \mid i \in [1, D]\}$ to one for $\{L'_i, R'_i \mid i \in [1, D]\}$. For each $i \in [1, D]$ s.t. $L_i < L'_i$, it updates the partial secret-key $\tilde{sk}_{L_i}$ for L_i to one for L'_i by $sk_{L'_i} \leftarrow \texttt{KUpd}_0(\tilde{sk}_{L_i}, L_i, L'_i, i)$. For each $i \in [1, D]$ s.t. $R_i > R'_i$, it updates $\tilde{sk}_{R_i}$ for R_i to one for R'_i by $sk_{R'_i} \leftarrow \texttt{KUpd}_1(\tilde{sk}_{R_i}, \hat{R}_i, \hat{R}'_i, i)$, where $\hat{R}'_i := T_i - 1 - R'_i$.

Finally, it returns $sk' := (\{sk_{L'_i}, sk_{R'_i} \mid i \in [1, D]\}, d)$.

$\texttt{Sig}(sk, l_1, r_1, \cdots, l_D, r_D, m)$**:** It parses sk for $\{L_i, R_i \mid i \in [1, D]\}$ and d as $(\{sk_{L_i}, sk_{R_i} \mid i \in [1, D]\}, d)$. It returns $\perp$ if $\neg[\bigwedge_{i \in [1, D]} 0 \leq l_i \leq r_i \leq T_i - 1 \bigwedge \sum_{\substack{i \in [1, D] \text{ s.t.} \\ L_i \leq l_i \leq r_i \leq R_i}} 1 \geq d]$.

Let $\mathbb{I}$ be a set $\{i \in [1, D]$ s.t. $L_i \leq l_i \leq r_i \leq R_i\}$ satisfying $|\mathbb{I}| \geq d$. For each $i \in \mathbb{I}$, let $(L'_i, R'_i) := (l_i, r_i)$. For each $i \in [1, D] \setminus \mathbb{I}$, let $(L'_i, R'_i) := (L_i, R_i)$.

Firstly, from sk, it generates a delegated and re-randomized secret-key for $\{L'_i, R'_i \mid i \in [1, D]\}$. Thus, $\tilde{sk} \leftarrow \texttt{KDel}(sk, \{L'_i, R'_i \mid i \in [1, D]\})$. $\tilde{sk}$ is parsed as $(\{sk_{L'_i}, sk_{R'_i} \mid i \in [1, D]\})$.

For every $i \in \mathbb{I}$, it does:

- Parse $sk_{L'_i}$ as $(D_{\log T_i,i}, d_{0,i}, \cdots, d_{\log T_i - 1,i}, \cdots)$.
- Parse $sk_{R'_i}$ as $(E_{\log T_i,i}, e_{0,i}, \cdots, e_{\log T_i - 1,i}, \cdots)$.
- Define a function $\mathit{\Delta}_{i,\mathbb{I}}(x) := \prod_{j \in \mathbb{I} \setminus \{i\}} \frac{x-j}{i-j}$.

For every $i \in [1, D] \setminus \mathbb{I}$, it does:

- $\hat{L}_i := T_i - 1 - L_i$.
- For every $j \in [0, \log T_i - 1]$, $s^*_{j,i}, t^*_{j,i} \xleftarrow{\mathrm{U}} \mathbb{Z}_p$.

It chooses $r \xleftarrow{\mathrm{U}} \mathbb{Z}_p$. It returns $\sigma := (U, \{V_{ji}, V'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, W)$, where

- $U := \prod_{i \in \mathbb{I}} (D_{\log T_i,i} \cdot E_{\log T_i,i})^{\mathit{\Delta}_{i,\mathbb{I}}(0)}$
  $\cdot \prod_{i \in [1,D] \setminus \mathbb{I}} \prod_{j \in [0, \log T_i - 1]} (u_{ji} v_0^{L_i[j]})^{s^*_{ji}} (w_{ji} v_0^{\hat{R}_i[j]})^{t^*_{ji}} \cdot (u \prod_{j \in [0,N-1]} v_j^{m[j]})^r$,
- $V_{ji} := (d_{ji})^{\mathit{\Delta}_{i,\mathbb{I}}(0)}$ and $V'_{ji} := (e_{ji})^{\mathit{\Delta}_{i,\mathbb{I}}(0)}$ (for $i \in \mathbb{I}$ and $j \in [0, \log T_i - 1]$),
- $V_{ji} := g^{s^*_{ji}}$ and $V'_{ji} := g^{t^*_{ji}}$ (for $i \in [1, D] \setminus \mathbb{I}$ and $j \in [0, \log T_i - 1]$),
- $W := g^r$.

$\mathtt{Ver}(\sigma, l_1, r_1, \cdots, l_D, r_D, m)$**:** It parses σ as $(U, \{V_{ji}, V'_{ji} \mid i \in [1, D], j \in [0, \log T_i - 1]\}, W)$. It returns 1 if it holds that

$$e(U, \tilde{g}) = e(g_1, g_2) \cdot \prod_{i \in [1,D]} \prod_{j \in [0, \log T_i - 1]} e(V_{ji}, \tilde{u}_{ji} \tilde{v}_0^{L_i[j]}) e(V'_{ji}, \tilde{w}_{ji} \tilde{v}_0^{\hat{R}_i[j]})$$
$$\cdot e(W, \tilde{u} \prod_{j \in [0,N-1]} \tilde{v}_j^{m[j]}).$$

It returns 0 otherwise.