Partially Non-Interactive Two-Round Lattice-Based Threshold Signatures

Rutchathon Chairattana-Apirom, Stefano Tessaro, and Chenzhi Zhu

Paul G. Allen School of Computer Science & Engineering
University of Washington, Seattle, USA
{rchairat,tessaro,zhucz20}@cs.washington.edu

Abstract. This paper gives the first lattice-based two-round threshold signature based on standard lattice assumptions for which the first message is independent of the message being signed without relying on fully-homomorphic encryption, and our construction supports arbitrary thresholds.
Our construction provides a careful instantiation of a generic threshold signature construction by Tessaro and Zhu (EUROCRYPT '23) based on specific linear hash functions, which in turns can be seen as a generalization of the FROST scheme by Komlo and Goldberg (SAC '20). Our reduction techniques are new in the context of lattice-based cryptography. Also, our scheme does not use any heavy tools, such as NIZKs or homomorphic trapdoor commitments.

1 Introduction

Multiple novel applications, primarily motivated by blockchains (e.g., digital wallets [GGN16]), are re-energizing a multi-decade agenda aimed at developing practical threshold signatures [Des88, DF90] with the goal of reducing trust assumptions in systems using digital signatures. To this end, recall that in a *t-out-of-n threshold signature scheme*, a set of n signers each hold shares of a secret signing key associated with a public verification key. Any subset of at least t of these signers should be able to come together and run a signing protocol to produce a signature on any message. However, an adversary that controls an arbitrary subset of fewer than t signers should not be able, on its own, to come up with a valid signature, even when they maliciously deviate from the protocol.

Threshold signatures are currently the focus of standardization efforts by NIST [Natnt] and IETF [CKGW22], and threshold signing protocols for a number of existing signature schemes have been given from a variety of cryptographic assumptions. These include threshold versions of BLS [Bol03, BL22], Schnorr [SS01, GJKR03, KG20, Lin24, BCK+22, CGRS23, CKM23a] and (EC-)DSA [GJKR96, GJKR07, GGN16, BGG19, GG18, LNR18, CGG+20], along with several schemes for ad-hoc signatures in pairing-free groups with specific properties [CKM+23b, TZ23, BLT+24]. Several RSA-based constructions [DDFY94, GRJK00, Sho00, DK01, TZ23] have also been proposed.

<u>Lattice-based threshold signatures.</u> With the threat of quantum computers looming on the horizon (and, in particular, their ability to break all assumptions behind all aforementioned threshold signatures), a widely recognized goal is to develop threshold signatures that are based on quantum-safe assumptions. The most natural candidate for such schemes are lattice-based assumptions, considering in particular the fact that NIST has selected DILITHIUM [LDK+22] and FALCON [PFH+22], two lattice-based signature schemes, for standardization. Regardless of quantum safety, it is also important to obtain constructions from a set of assumptions as diverse as possible.

While lattice-based cryptography has been enormously successful in enabling extremely sophisticated functionalities, building efficient lattice-based threshold signatures has turned out to be

Scheme	Offline	Online	Assumption	Notes
ASY [ASY22]	0	1	SIS + LWE	FHE
GKS [GKS24]	0	2	RSIS + RLWE	Trapdoor Commitments + NIZKs
TRaccoon [PKM+24]	0	3	MSIS + MLWE	
KRT [KRT24]	2	3	MSIS + MLWE	Adaptive security
Pelican [ENP24]	0	4	MLWE	Robustness when $t \leqslant n/3$
EKT [EKT24]	1	1	AOM-MLWE	Non-standard assumption
Our work	1	1	MSIS	

Table 1. Overview of lattice-based t-out-of-n threshold signatures for arbitrary thresholds with the number of online (message-dependent) and offline (message-independent) rounds, hardness assumptions, and notes on specific details in each construction.

---

very challenging. In principle, the problem can be solved generically and round optimally with constructions [BGGK17, BGG+18, ASY22] based on *Fully-Homomorphic Encryption* (FHE), but these require the homomorphic evaluation of the signing algorithm *within* the FHE, thus imposing a substantial computational and communication overhead on the signing process.

There have been attempts [DOTT21, Che23] at giving more direct constructions of two-round signing protocols based on the Fiat-Shamir-with-abort paradigm [Lyu09], obtained by adapting constructions for the related notion of multi-signatures. These constructions only realize n-out-of-n threshold signatures, i.e., do not tolerate arbitrary thresholds $t < n$. Gur, Katz, and Silde [GKS24] recently proposed a new two-round construction based on linearly homomorphic encryption (LHE) which supports arbitrary thresholds. Both rounds are message-dependent, and they rely on homomorphic trapdoor commitments and NIZKs to ensure security against malicious signers. For $n = 5$ and $t = 3$, their signatures and public keys have sizes 46.6 and 13.6 KB, respectively, whereas the communication costs for signing are roughly 3 MB per signer. Recent work by del Pino *et al.* [PKM+24] proposes a more efficient lattice-based threshold signature scheme that does not rely on FHE or the aforementioned heavy primitives, but the drawback is that the protocol has three message-dependent rounds. Other recent works consider adaptive security [KRT24] and robustness [ENP24], but their schemes require higher round complexity. In Table 1, we provide an overview of the aforementioned lattice-based threshold signatures, detailing round complexity and assumptions used in each construction. We further discuss a concurrent and independent work by Espitau et al. [EKT24] below.

<u>Better two-round threshold signatures.</u> In this paper, we pursue the question of designing better and more efficient two-round threshold signatures. Clearly, we would like to minimize communication along with signature and key sizes, but other properties are desirable. For example, a fundamental property of FROST [KG20, BCK+22] is that it is *partially* non-interactive, in that while the signing protocol consists of two rounds, the first round messages are simply nonces *independent* of the message being signed. This allows us to recover some of the positive features of non-interactive schemes by preprocessing the initial round. Currently, with the exception of FHE-based schemes, we do not know of any partially non-interactive lattice-based threshold signatures. Note that in fact partially non-interactive lattice-based multi-signatures exist [BTT22], inspired by the discrete-log based counterparts [NRS21], but it is not clear how to turn these into threshold signatures, especially for the case $t < n$.

<u>Our contributions.</u> In this paper, we develop the first partially non-interactive lattice-based threshold signatures that do not rely on FHE or other heavy primitives like NIZKs and trapdoor

commitments. The security of our scheme is based on standard lattice assumptions, in particular, we rely on the Module-SIS assumption.

To achieve 128-bit of security and allow for up to 2^{64} signatures to be generated with the same key, for the case $n = 5$, which is the same setting considered by [GKS24], the signatures in our scheme have sizes roughly of 258.1 KB with the size of public keys 47.6 KB, and the communication complexity per signer is 1.5 MB. While our signature and public key sizes are larger than [GKS24], we achieve better communication complexity.

Like other recent works [BCK+22, BLT+24, CKM23a, PKM+24], we do not propose an explicit distributed key generation (DKG) protocol. (We can envision that keys are either set up manually, or that they are the output of a suitable generic MPC protocol.) We leave the design of suitable DKG protocols as an interesting open question.

<u>Our approach.</u> A common way to construct an efficient lattice-based primitive is to take an efficient construction based on pairing-free groups and translate it into a lattice-based scheme. However, one key barrier in translating ideas from FROST, the state-of-art group-based partially non-interactive threshold signature scheme, to the lattice setting is that the security analysis of FROST relies on the one-more discrete logarithm assumption, of which no analog is known in the lattice setting. A recent work by Tessaro and Zhu [TZ23] proposes a variant of FROST based on linear hash functions (LHF) and gives a security reduction to the plain DL assumption. Inspired by the work of Hauck et al. [HKLN20], which turns a LHF-based blind signature scheme into a lattice-based one, our starting point is to translate the LHF-based threshold signatures into lattice-based threshold signatures. The main difficulty in this idea is that the lattice-based linear hash functions do not have the desirable algebraic properties as required in the original analysis from [TZ23]. We refer to the technical overview below for the detailed issues and our solutions.

We also want to point out some caveats with our work:

- Our approach is very different from the aforementioned lattice-based threshold signature works, and techniques, such as modularising the proof by reducing to the self-target MSIS problem [DKL+18, KLS18] or the one from [Lyu12] for reducing the norm bound of secret keys, do not apply to our construction. We refer to the last paragraph of the technical overview for more details.
- Our solution requires stronger properties from the underlying secret sharing scheme, which are satisfied by the secret sharing scheme by Benaloh and Leichter [BL90]. However, this makes our secret key shares significantly large and we address this further in Section 4.5. We remark that Benaloh-Leichter and similar kinds of small-coefficient linear secret sharing have also been used before in the context of lattice-based threshold cryptography, e.g., in constructing universal thresholdizers [BGGK17, BGG+18, BS23, CCK23]. We further discuss in Section 3.3 the issues which make other secret sharing schemes, such as the one by Applebaum et al. [ANP23], not applicable to our use case.

<u>Significance of the work.</u> We emphasize that we see the primary value of our paper in showing the feasibility of constructing partially non-interactive threshold signatures based on standard lattice assumptions without using FHE and giving new techniques involved in transforming a DL-based schemes into a lattice-based one. Nonetheless, we note that the efficiency of our schemes is still within the practical realm and deserves further investigation.

<u>Concurrent and independent work.</u> Espitau et al. [EKT24] recently proposed a lattice-based two-round partially non-interactive threshold signatures. Their construction also follows the approach of instantiating FROST in the lattice-based settings. However, the difference is that their

security analysis is based on a non-standard interactive assumption, the Algebraic One-More Module Learning with Error (AOM-MLWE) assumption, which is newly proposed in their paper.

<u>Other related works.</u> We discuss some additional related works we have not discussed above. An alternative approach to obtain threshold signatures is to leverage standard MPC techniques to evaluate (part of the) signing. For example, Bendlin *et al.* [BKP13] use this approach to obtain a threshold version of GPV signatures [GPV08]. More recently, Cozzo and Smart [CS19] considered more broadly MPC-based instantiations of NIST post-quantum signature candidates and concluded that they are unlikely to lead to practical solutions.

1.1 Technical Overview

Our starting point is a variant of FROST [KG20] proposed by [TZ23] which gives a threshold signature scheme based solely on the DL assumption, instead of the stronger one-more DL assumption. The key idea is to replace the map $x \mapsto g^x$ (for a generator g) in FROST with a *compressing* and *collision resistant linear* map $F : \mathfrak{D} \to \mathfrak{R}$, referred to as a linear hash function (LHF), where $\mathfrak{D}$ and $\mathfrak{R}$ are two vector spaces over a scalar field $\mathfrak{S}$. The secret key of the scheme is a random element $\mathsf{sk} \in \mathfrak{D}$ and the corresponding public key is $\mathsf{pk} \leftarrow F(\mathsf{sk})$. The secret key shares $\{\mathsf{sk}_i\}_{i\in[n]}$ are generated using Shamir's secret sharing. The signing protocol consists of one offline round and one online round.

- In the offline round, each signer i samples $r_{i,0}, r_{i,1} \in \mathfrak{D}$ and publishes a token $(R_{i,0}, R_{i,1}) \leftarrow (F(r_{i,0}), F(r_{i,1}))$.
- In the online round, to sign a message μ, the user selects a set of signers SS of size at least t and sends a request $lr \leftarrow (\mu, SS, \{R_{i,0}, R_{i,1}\}_{i\in SS})$ to each signer in SS. Each signer i sends $R \leftarrow \sum_{i'\in SS}(R_{i',0} + bR_{i',1})$ with $b \leftarrow \mathrm{H}_1(\mathsf{pk}, lr)$, and $z_i \leftarrow r_{i,0} + br_{i,1} + c\lambda_i^{SS}\mathsf{sk}_i$ with $c \leftarrow \mathrm{H}_2(\mathsf{pk}, \mu, R)$ to the user, where H_1 and H_2 are two hash functions.
- Finally, the signature is computed as $(R, z = \sum_{i\in SS} z_i)$. To verify it, one checks whether $F(z) = R + c \cdot \mathsf{pk}$ for $c = \mathrm{H}_2(\mathsf{pk}, \mu, R)$.

Here $\mathrm{H}_1, \mathrm{H}_2 : \{0,1\}^* \to \mathfrak{S}$ are hash functions. We note that the underlying signature scheme can be viewed as a LHF-based analog of Schnorr signatures. Also, the required properties of F are:

(i) Linearity: $F(a) + F(b) = F(a + b)$ holds for any $a, b \in \mathfrak{D}$.
(ii) Collision resistance: it is hard to find $x \neq y \in \mathfrak{D}$ such that $F(x) = F(y)$ for a randomly sampled F.
(iii) Compressing: the pre-image of any element in $\mathfrak{R}$ under F contains multiple elements.

As observed by Hauck et al. [HKLN20], a natural candidate to instantiate LHF from lattices is $F(\boldsymbol{x}) = A\boldsymbol{x}$, where A is a uniformly random matrix $A \in R_q^{k\times m}$ for a prime q and the ring $R_q := \mathbb{Z}_q[X]/(X^N + 1)$, with $\mathfrak{D} = \{\boldsymbol{x} \in R_q^m \mid \|\boldsymbol{x}\|_\infty \leqslant \beta_x\}$, $\mathfrak{R} = R_q^k$, and $\mathfrak{S} = R_q$, where $\beta_x < q$ is a constant. It is clear that F is linear and compressing if $|\mathfrak{D}| = (2\beta_x)^{mN} \gg q^{kN} = |\mathfrak{R}|$. Also, F is collision resistance under the Module-SIS (MSIS) assumption, which guarantees that given a uniform matrix $A \in R_q^{k\times m}$, it must be infeasible to find a small-norm solution $\boldsymbol{x} \neq \boldsymbol{0}$ such that $A\boldsymbol{x} = \boldsymbol{0}$. If one can find $\boldsymbol{x}_1 \neq \boldsymbol{x}_2 \in \mathfrak{D}$ such that $F(\boldsymbol{x}_1) = F(\boldsymbol{x}_2)$, we have $A(\boldsymbol{x}_1 - \boldsymbol{x}_2) = 0$, which gives us a MSIS solution $(\boldsymbol{x}_1 - \boldsymbol{x}_2)$ for A with ℓ_∞-norm bounded by $2\beta_x$.

Unfortunately, we cannot simply apply the analysis from [TZ23] to the above lattice-based instantiation. A simple reason is that $\mathfrak{D}$ as defined above is not a linear space,[1] which are required

[1] This is because given $\boldsymbol{x}_1, \boldsymbol{x}_2$ with ℓ_∞-norm bounded by β_x, $\|\boldsymbol{x}_1 + \boldsymbol{x}_2\|_\infty$ can exceed β_x.

by the prior analysis. There are also more technical reasons why this does not work, and to see what they are, we now try to apply the prior analysis here.

<u>Reduction idea from prior work.</u> The reduction idea is simple. Denote an adversary that breaks unforgeability of the threshold signature scheme as $\mathcal{A}$, which corrupts up to $t-1$ signers, engages in an arbitrary number of signing sessions with honest signers, and forges a valid signature for a message that was not signed in any of the signing sessions. We construct a MSIS adversary $\mathcal{B}$ as follows: (In the analysis, H_1 and H_2 are modeled as random oracles.) Initially, $\mathcal{B}$ receives a MSIS challenge A. Then, $\mathcal{B}$ runs $\mathcal{A}$ by simulating the key generation, the signing sessions and the random oracles following the protocol by itself. If $\mathcal{A}$ returns a valid message-signature pair $(\mu^*, sig^* = (\boldsymbol{R}^*, \boldsymbol{z}^*))$, $\mathcal{B}$ rewinds $\mathcal{A}$ to the step that the query $\mathrm{H}_2(\mathsf{pk}, \mu^*, \boldsymbol{R}^*)$ is made and runs $\mathcal{A}$ again while answering its random oracle queries with refreshed randomness. If $\mathcal{A}$ returns $(\bar{\mu}^*, \overline{sig}^* = (\overline{\boldsymbol{R}}^*, \bar{\boldsymbol{z}}^*))$[2] with $(\mu^*, \boldsymbol{R}^*) = (\bar{\mu}^*, \overline{\boldsymbol{R}}^*)$, then we find a collision $F(\boldsymbol{z}^* - c \cdot \mathsf{sk}) = \boldsymbol{R}^* = \overline{\boldsymbol{R}}^* = F(\bar{\boldsymbol{z}}^* - \bar{c} \cdot \mathsf{sk})$, where c and $\bar{c}$ are the outputs of $\mathrm{H}_2(\mathsf{pk}, \mu^*, \boldsymbol{R}^*)$ in the first and second execution respectively. Therefore, $\mathcal{B}$ returns $(\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - (c - \bar{c}) \cdot \mathsf{sk})$. Otherwise, $\mathcal{B}$ aborts.

By the Forking Lemma, if $\mathcal{A}$ breaks unforgeability with high probability, then we have that $\mathcal{B}$ does not abort and $c \neq \bar{c}$ with high probability. The difficulty here is to ensure that we indeed find a MSIS solution, i.e. $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - (c - \bar{c}) \cdot \mathsf{sk} \neq 0$. The prior analysis from [TZ23] shows that for any two secret keys $\mathsf{sk} \neq \mathsf{sk}'$ mapping to the same public key, there exists a bijection Φ that maps the randomness ρ of $\mathcal{B}$ to another randomness ρ' such that ρ and ρ' lead to secret key sk and sk' respectively, and the view of $\mathcal{A}$ given ρ is identical to that given ρ'. Therefore, $\mathcal{A}$ outputs the same $(\mu^*, \boldsymbol{R}^*, \boldsymbol{z}^*, \bar{\mu}^*, \overline{\boldsymbol{R}}^*, \bar{\boldsymbol{z}}^*)$ independent of whether $\mathcal{B}$ is run with ρ or ρ'. Since $\mathsf{sk} \neq \mathsf{sk}'$ and $c \neq \bar{c}$, we have that $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - (c - \bar{c}) \cdot \mathsf{sk} \neq \boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - (c - \bar{c}) \cdot \mathsf{sk}'$, so $\mathcal{B}$ wins in at least one of the cases. Hence, $\mathcal{B}$ wins with at least half of the probability that $\mathcal{B}$ does not abort.

<u>Challenges in lattice instantiations.</u> The main challenges lie in how to construct Φ. Note that given the secret key sk, the randomness ρ of $\mathcal{B}$ consists of: an RO tape $\boldsymbol{h} = (h_{1,1}, h_{1,2}, \ldots, h_{\mathsf{q}_h,1}, h_{\mathsf{q}_h,2}, \bar{h}_{1,1}, \ldots, \bar{h}_{\mathsf{q}_h,2})$, where $h_{i,j}$ is used to answer the i-th RO query to H_j in the first execution of $\mathcal{A}$ and $\bar{h}_{i,j}$ is used after rewinding, the secret key shares $\{\mathsf{sk}_i\}_{i \in [n]}$ of sk,[3] and the randomness $(\boldsymbol{r}_{i,0}, \boldsymbol{r}_{i,1})$ for generating the tokens of each signing session. Therefore, we only consider Φ defined over those variables. First of all, Φ maps $\boldsymbol{h}$ to itself since $\mathcal{A}$ can learn $\boldsymbol{h}$ from RO queries. For the other two parts, Φ satisfies the following:

(1) Φ maps $\{\mathsf{sk}_i\}_{i \in [n]}$ to $\{\mathsf{sk}'_i\}_{i \in [n]}$ such that $\{\mathsf{sk}'_i\}_{i \in [n]}$ are the secret shares of sk' and $\mathsf{sk}_i = \mathsf{sk}'_i$ for any corrupted signer i.

(2) For the interaction with signer i during signing, Φ maps $(\boldsymbol{r}_{i,0}, \boldsymbol{r}_{i,1})$ to $(\boldsymbol{r}'_{i,0}, \boldsymbol{r}'_{i,1})$ such that $F(\boldsymbol{r}_{i,0}) = F(\boldsymbol{r}'_{i,0})$, $F(\boldsymbol{r}_{i,1}) = F(\boldsymbol{r}'_{i,1})$, and

$$\begin{pmatrix} 1 & b \\ 1 & \bar{b} \end{pmatrix} \begin{pmatrix} \boldsymbol{r}_{i,0} \\ \boldsymbol{r}_{i,1} \end{pmatrix} + \begin{pmatrix} c\lambda_i^{SS} \mathsf{sk}_i \\ \bar{c}\lambda_i^{SS} \mathsf{sk}_i \end{pmatrix} = \begin{pmatrix} \boldsymbol{z}_i \\ \bar{\boldsymbol{z}}_i \end{pmatrix} = \begin{pmatrix} 1 & b \\ 1 & \bar{b} \end{pmatrix} \begin{pmatrix} \boldsymbol{r}'_{i,0} \\ \boldsymbol{r}'_{i,1} \end{pmatrix} + \begin{pmatrix} c\lambda_i^{SS} \mathsf{sk}'_i \\ \bar{c}\lambda_i^{SS} \mathsf{sk}'_i \end{pmatrix},$$

where we use $\bar{(\cdot)}$ to denote the variables after rewinding. (It is possible that the adversary makes only one query or the same queries for the token during the two executions, but these cases are easier to deal with. Thus, we only discuss the above hardest case here.)

---

[2] In this section, we will use the overline to denote values in the second run.

[3] More accurately, it should be the randomness for generating the secret key shares, but for simplicity of explanation, we use the secret key shares instead.

It is not hard to satisfy the first condition due to the privacy property of secret sharing. For the second condition, by the idea of prior work, if $b - \bar{b}$ is invertible, we can set $(\boldsymbol{r}'_{i,0}, \boldsymbol{r}'_{i,1}) = (\boldsymbol{r}_{i,0} + (c - b(b-\bar{b})^{-1}\Delta_c)\Delta_{\mathsf{sk}}, \boldsymbol{r}_{i,1} + (b-\bar{b})^{-1}\Delta_c\Delta_{\mathsf{sk}})$ to make the condition hold, where $\Delta_c = c - \bar{c}$ and $\Delta_{\mathsf{sk}} = \lambda_i^{SS}(\mathsf{sk}_i - \mathsf{sk}'_i)$. However, the problem is that the map is not a bijection since $\mathfrak{D}$ is not a vector space. There is no guarantee that $(\boldsymbol{r}'_{i,0}, \boldsymbol{r}'_{i,1}) \in \mathfrak{D}$ for $\boldsymbol{r}_{i,0}, \boldsymbol{r}_{i,1} \in \mathfrak{D}$. A common solution, which was also used by Hauck et al. [HKLN20], is to enlarge $\mathfrak{D}$ (by increasing β_x) such that $(\boldsymbol{r}'_{i,0}, \dots, \boldsymbol{r}'_{i,\ell}) \in \mathfrak{D}$ except for a negligible fraction of $(\boldsymbol{r}_{i,0}, \dots, \boldsymbol{r}_{i,\ell})$. Still, there are two issues we need to address: (a) We need to show that the shift $(b-\bar{b})^{-1}\Delta_c\Delta_{\mathsf{sk}}$ is small; (b) To make the fraction of bad randomness negligible, we have to set $\beta_x = \Omega(2^\kappa \left\|(b-\bar{b})^{-1}\Delta_c\Delta_{\mathsf{sk}}\right\|)$, where κ denotes the security parameter. This would lead to a very large modulus.

<u>Our solution.</u> For issue (a), we need to show that all of the three parts, i.e., $(b-\bar{b})^{-1}$, Δ_c, and Δ_{sk}, are small. To make sure that the inverse of $(b-\bar{b})^{-1}$ is small, we restrict the range of H_1 to be $\{0,1\}$. As a result, with 1/2 probability, $b - \bar{b} \in \{1, -1\}$ and thus its inverse is small (either 1 or -1). Then, we boost the probability to $1 - 2^{-2\kappa}$ by increasing the number of nonces and the range of H_1 to be $\{0,1\}^{2\kappa}$. More precisely, in the offline round, each signer i samples $\boldsymbol{r}_{i,0}, \boldsymbol{r}_{i,1}, \dots, \boldsymbol{r}_{i,\ell}$ for $\ell = 2\kappa$. In the online round, signer i returns $\boldsymbol{z}_i \leftarrow \boldsymbol{r}_{i,0} + \sum_{j\in[\ell]} b_j \boldsymbol{r}_{i,j} + c\lambda_i^{SS}\mathsf{sk}_i$, where $(b_1, \dots, b_\ell) \in \{0,1\}^\ell$ are computed from H_1. Also, Φ maps $(\boldsymbol{r}_{i,0}, \dots, \boldsymbol{r}_{i,\ell})$ to $(\boldsymbol{r}'_{i,0}, \dots, \boldsymbol{r}'_{i,\ell}) = (\boldsymbol{r}_{i,0} + (c - b_j(b_j - \bar{b}_j)^{-1}\Delta_c)\Delta_{\mathsf{sk}}, \dots, \boldsymbol{r}_{i,j-1}, \boldsymbol{r}_{i,j} + (b_j - \bar{b}_j)^{-1}\Delta_c\Delta_{\mathsf{sk}}, \boldsymbol{r}_{i,j+1}, \dots, \boldsymbol{r}_{i,\ell})$, where j is the first index with $b_j \neq \bar{b}_j$.

For Δ_c, it is a common practice to sample c with small ℓ_1-norm, which implies that the norm of Δ_c is small. Lastly, we have to ensure that the norm of Δ_{sk} is small. This imposes an additional requirement on the secret sharing scheme. Namely, it requires that there exists a map Φ satisfying the aforementioned condition (1) and in addition, restricting $\left\|\mathsf{sk}_i - \mathsf{sk}'_i\right\|_\infty$ to be small. We show that a special class of secret sharing schemes, referred to as linear secret sharing schemes with small coefficients, satisfies the requirement. We refer to Section 3 for the detailed definition and instantiation.

To address issue (b), we sample each $\boldsymbol{r}_{i,j}$ from an m-dimensional discrete Gaussian distribution centered at the origin with variance σ_{r}. Intuitively, $\mathfrak{D}$ becomes a probability distribution instead of a set, and we can show that $\mathcal{B}$ wins with high probability as long as the ratio $\alpha = \frac{\Pr[(\boldsymbol{r}_{i,0},\dots,\boldsymbol{r}_{i,\ell})]}{\Pr[\Phi(\boldsymbol{r}_{i,0},\dots,\boldsymbol{r}_{i,\ell})]}$ is close to 1 except for a negligible fraction of $(\boldsymbol{r}_{i,0}, \dots, \boldsymbol{r}_{i,\ell})$. More precisely, we need to show $\alpha^{\mathsf{q}_s} \in (1-\varepsilon, 1+\varepsilon)$ for some constant ε, where q_s denotes the number of signing sessions. Since the map only shifts $\boldsymbol{r}_{i,0}$ and $\boldsymbol{r}_{i,j}$ by roughly $\Delta = \Delta_c\Delta_{\mathsf{sk}}$, the ratio is roughly $\exp\left(\frac{\|\Delta\|^2 + 2\|\Delta\| \cdot \left\|(\boldsymbol{r}_{i,0},\dots,\boldsymbol{r}_{i,\ell})\right\|}{\sigma_{\mathsf{r}}^2}\right)$, and we can achieve the desired bound by setting $\sigma_{\mathsf{r}} = \Omega(\mathsf{q}_s \|\Delta_c\Delta_{\mathsf{sk}}\|)$.

We now discuss the two optimizations made to improve the efficiency of our protocol in the following paragraphs.

<u>Decreasing the number of nonces.</u> In the above protocol, the number of nonces generated is equal to the security parameter, resulting in significant overhead in communication complexity. To decrease the number of nonces ℓ, the key observation is that we can extend the domain of b to the set of signed monomials, $\mathcal{S}_{\mathsf{b}} := \{\pm 1, \pm X, \dots, \pm X^{N-1}\} \subseteq R_q$. (This has also been considered in other works, e.g., [BCK+14].) Although for any $b \neq \bar{b} \in \mathcal{S}_{\mathsf{b}}$, $(b - \bar{b})$ does not necessarily have a small inverse, we can show that there exists $v_{b-\bar{b}} \in R$ such that $v_{b-\bar{b}}(b-\bar{b}) = 2$ and $\left\|v_{b-\bar{b}}\right\|_\infty \leqslant 1$. Therefore, we let each signer compute $\boldsymbol{z}_i \leftarrow \boldsymbol{r}_{i,0} + \sum_{j\in[\ell]} b_j \boldsymbol{r}_{i,j} + 2c \cdot \lambda_i^{SS}\mathsf{sk}_i$, and, then we can structure the map Φ following the method described above except that we replace $(b-\bar{b})^{-1}$ with

$v_{b-\bar{b}}$. As a result, we just need to set $\ell = 2\kappa/\log(2N)$, which is 10 times smaller for $N = 512$ used in our concrete efficiency analysis.

<u>Improving modulus size using the Rényi Divergence.</u> Another main efficiency problem is that the modulus size depends linearly on q_s, which is implied by how we set σ_r. To address this, we observe that the ratio $\frac{\Pr[(\boldsymbol{r}_{i,0},\ldots,\boldsymbol{r}_{i,\ell})]}{\Pr[(\boldsymbol{r}'_{i,0},\ldots,\boldsymbol{r}'_{i,\ell})]}$ is not evenly distributed. It gets larger as the norm of $\boldsymbol{r}_{i,j}$ becomes larger. However, as the norm of $\boldsymbol{r}_{i,j}$ becomes larger, its probability of being sampled becomes exponentially small. As a result, there are only a small fraction of points with ratios close to the ratio bound, while a large proportion of points have much smaller ratios. Therefore, we try to use the Rényi divergence, which computes the *average* of the probability ratio of two distributions. More precisely, instead of considering the probability that a particular random value $(\boldsymbol{r}_{i,0}, \ldots, \boldsymbol{r}_{i,\ell})$ is sampled, we consider the distribution of the view of $\mathcal{A}$ conditioning on sk (denoted by T_sk) directly. We show that $\mathcal{B}$ wins with high probability as long as the Rényi divergence $R_\alpha(T_\mathsf{sk} \| T_{\mathsf{sk}'})$ is close to 1. Then, we observe that the Rényi divergence of the view of $\mathcal{A}$ in a *single* signing session given sk from that given sk' is roughly the Rényi divergence of two discrete Gaussian distributions both with variance $O(\sigma_\mathsf{r})$ and with distance $\|\Delta_c\Delta_\mathsf{sk}\|$ between their centers. Thus, considering all signing sessions (both before and after the rewinding), $R_\alpha(T_\mathsf{sk} \| T_{\mathsf{sk}'})$ is roughly $\exp\left(\mathsf{q}_s \|\Delta_c\Delta_\mathsf{sk}\|^2/\sigma_\mathsf{r}^2\right)$, omitting the constants and unimportant factors. Therefore, we can set $\sigma_\mathsf{r} = \Omega(\sqrt{\mathsf{q}_s}\|\Delta_c\Delta_\mathsf{sk}\|)$, improving the modulus size by a factor of $\sqrt{\mathsf{q}_s}$. We note that similar techniques have also been used by Agrawal et al. [ASY22] to improve the modulus size of their FHE-based threshold signature.

<u>Technical distinctions from other works.</u> We emphasize that our proof framework is very different from other recent lattice-based threshold and multi-signature works [DOTT21, BTT22, Che23, PKM+24, EKT24] without using homomorphic encryptions, where unforgeability of the protocols is reduced to the key-only security of the underlying signature schemes by showing that the signing oracles can be simulated given only the public key. In contrast, our proof framework directly simulates the unforgeability game using the honestly sampled secret key and then extract an MSIS solution by rewinding. The benefit here is that we do not need to rely on more advanced assumptions [EKT24], introduce an additional online round [Che23, PKM+24], or use hard cryptographic primitives, such as a homomorphic trapdoor commitment scheme [DOTT21, BTT22]. However, the downside is that techniques, such as reducing to the self-target MSIS problem [DKL+18, KLS18] or the common trick to reduce the norm bound of secret keys using MLWE [Lyu12, DOTT21, BTT22], do not apply to our construction.

2 Preliminaries

2.1 Notations

For any integers $0 \leqslant k < n$, $[n]$ denotes $\{1, \ldots, n\}$, and $[k..n]$ denotes $\{k, \ldots, n\}$. We use κ to denote the security parameter. For a finite set S, $|S|$ denotes the size of S, and $x \leftarrow\!\!\$\ S$ denotes sampling an element uniformly from S and assigning it to x. For a distribution $\mathcal{D}$, $x \leftarrow\!\!\$\ \mathcal{D}$ denotes sampling x according to $\mathcal{D}$. For a sequence of variables $x_1, \ldots, x_\ell$, we use $x_{[i..j]}$ to denote $(x_i, \ldots, x_j)$. For any vector space V over a field F and a set $S \in V$, we denote $\mathsf{Span}_F(S)$ as the F-span of S, which is the smallest F-subspace of V that contains S. In particular, we omit F from the subscript if $F = \mathbb{R}$. For a finite set $S = \{v_1, \ldots, v_n\} \subseteq V$, we say S is F-linearly independent if and only if for any non-zero $(a_1, \ldots, a_n) \in F^n$, $\sum_{i\in[n]} a_i v_i \neq 0$. We say S is a F-basis of V if and only if S is F-linearly independent and $\mathsf{Span}_F(S) = V$. When F is not specified, we assume $F = \mathbb{R}$. The dimension of V is equal to the size of S.

2.2 Polynomial Rings

Let q be an odd prime and N be a power of 2. We denote the ring $R := \mathbb{Z}[X]/(X^N + 1)$, contained in the cyclotomic field $K := \mathbb{Q}[X]/(X^N + 1)$, and let $R_q := R/qR \cong \mathbb{Z}_q[X]/(X^N + 1)$. Denote $K_{\mathbb{R}} := \mathbb{R} \otimes K \cong \mathbb{R}[X]/(X^N + 1)$. For an element $v \in K_{\mathbb{R}}$, where $v = \sum_{i=0}^{N-1} v_i X^i$, we denote its conjugate as $v^* = \sum_{i=0}^{N-1} -v_i X^{N-i}$. We use ϕ to denote the coefficient embedding that embeds $K_{\mathbb{R}}$ in $\mathbb{R}^N$, and ϕ maps v to vector $(v_0, \ldots, v_{N-1}) \in \mathbb{R}^N$. When applying ϕ to a vector $\boldsymbol{v} \in K_{\mathbb{R}}^m$, ϕ maps $\boldsymbol{v}$ to a vector in $\mathbb{R}^{mN}$ by applying ϕ to each entry of $\boldsymbol{v}$. The map ϕ is a bijection, and we denote its inverse by ϕ^{-1}. An ℓ_p-norm of $\boldsymbol{v} \in K_{\mathbb{R}}^m$ is given by

$$\|\boldsymbol{v}\|_p := \|\phi(\boldsymbol{v})\|_p = \left(\sum_{i=1}^{m} \sum_{j=0}^{N-1} |v_{i,j}|^p \right)^{\frac{1}{p}} ,$$

where $v_{i,j}$ denotes the coefficient of X^j of the i-th entry of $\boldsymbol{v}$. Additionally, the ℓ_∞-norm of $\boldsymbol{v}$ is defined as $\|\boldsymbol{v}\|_\infty := \max_{i \in [m], j \in [0..N-1]} |v_{i,j}|$. For the ℓ_2-norm, we omit the subscript and denote $\|\boldsymbol{v}\|$ as the ℓ_2-norm of $\boldsymbol{v}$. Denote the conjugate transpose of $\boldsymbol{v} \in K_{\mathbb{R}}^m$ as $\boldsymbol{v}^\dagger := (\boldsymbol{v}^*)^T$. We define the inner product of two vectors $\boldsymbol{v}, \boldsymbol{v}' \in K_{\mathbb{R}}^m$ as $\langle \boldsymbol{v}, \boldsymbol{v}' \rangle := \phi(\boldsymbol{v})^T \phi(\boldsymbol{v}') = \langle \phi(\boldsymbol{v}), \phi(\boldsymbol{v}') \rangle$. We have $\|\boldsymbol{v}\|^2 = \langle \boldsymbol{v}, \boldsymbol{v} \rangle$. We say $\boldsymbol{v}$ is a unit vector if $\|\boldsymbol{v}\| = 1$.

Also, we define a map ϕ_{M} that maps each element in $K_{\mathbb{R}}$ to a matrix in $\mathbb{R}^{N \times N}$ as follows. Let $M_X := \begin{pmatrix} \mathbf{0} & -1 \\ \mathbb{I}_{N-1} & \mathbf{0} \end{pmatrix} \in \mathbb{R}^N$, where I_{N-1} is the identity matrix in $\mathbb{R}^{N-1}$. For each $v \in K_{\mathbb{R}}$, $\phi_{\mathsf{M}}(v) := \sum_{i=0}^{N-1} v_i M_X^i$, which can be viewed as the matrix representation of v. In particular, for ϕ and ϕ_{M}, the following properties hold: for any $v, v' \in K_{\mathbb{R}}$, $\phi_{\mathsf{M}}(v^*) = \phi_{\mathsf{M}}(v)^T$, $\phi_{\mathsf{M}}(vv') = \phi_{\mathsf{M}}(v)\phi_{\mathsf{M}}(v')$ and $\phi_{\mathsf{M}}(v)\phi(v') = \phi(vv')$. We extend the above definitions to R_q by representing each $v \in R_q$ as $v = \sum_{i=0}^{N-1} v_i X^i$, where $v_i \in \{-(q-1)/2, \ldots, (q-1)/2\}$.

For a matrix $M \in K_{\mathbb{R}}^{m \times m}$, we denote its conjugate transpose as $M^\dagger = (M^*)^T$, and we say M is *hermitian* if $M = M^\dagger$. We say M is *positive definite* if and only if M is hermitian and for all $\boldsymbol{x} \in K_{\mathbb{R}}^m \backslash \{\mathbf{0}\}$, $\langle \boldsymbol{x}, M\boldsymbol{x} \rangle > 0$, or equivalently, $\phi_{\mathsf{M}}(M)$ is positive definite. Also, denote $\sigma_{\min}(M) := \inf_{\boldsymbol{x} \in K_{\mathbb{R}}^m, \|\boldsymbol{x}\|=1} \langle \boldsymbol{x}, M\boldsymbol{x} \rangle$ as the smallest singular value of M and $\sigma_{\max}(M) := \sup_{\boldsymbol{x} \in K_{\mathbb{R}}^m, \|\boldsymbol{x}\|=1} \langle \boldsymbol{x}, M\boldsymbol{x} \rangle$ as the largest singular value of M.

We state the following lemma establishing the property of the set of signed monomials $\mathcal{S}_{\mathsf{b}} := \{\pm 1, \ldots, \pm X^{N-1}\} \subseteq R_q$, used in the security analysis.

Lemma 1. *Let $\mathcal{S}_{\mathsf{b}} := \{\pm 1, \ldots, \pm X^{N-1}\} \subseteq R_q$. For any $b, \bar{b} \in \mathcal{S}_{\mathsf{b}}$ such that $b \neq \bar{b}$, the ideal generated by $b - \bar{b}$ contains 2.*

Proof. Let $b = sX^a, \bar{b} = \bar{s}X^{\bar{a}}$ for $a, \bar{a} \in [0..N-1]$ and $s, \bar{s} \in \{-1, 1\}$. Consider two cases:

- $a = \bar{a}$: Then, $b - \bar{b} = 2X^a$ or $-2X^a$. It is easy to see that the statement holds as $(b - \bar{b})X^{N-a} = 2$ or -2.
- $a \neq \bar{a}$: W.l.o.g. assume $a > \bar{a}$. Then, $b - \bar{b}$ generates $X^{a-\bar{a}} - s\bar{s}$ since $(b - \bar{b}) \cdot (-sX^{N-\bar{a}}) = X^{a-\bar{a}} + s\bar{s}$. We can see that this generates $X^{2^e(a-\bar{a})} - 1$ for any $e \geqslant 1$, since $(X-1)(X+1) = X^2 - 1$. Since $a - \bar{a} < N$ and N is a power of 2, there exists e such that $N | 2^e(a - \bar{a})$ but $N \nmid 2^{e-1}(a - \bar{a})$. Then, $2^e(a - \bar{a}) = Na'$ for some odd a', and thus $b - \bar{b}$ generates $X^{Na'} - 1 = (-1)^{a'} - 1 = -2$, which implies the statement. □

2.3 Lattices and Discrete Gaussian Distributions

In this subsection, we give definitions for lattices and discrete Gaussian distributions over $\mathbb{R}$ and $K_{\mathbb{R}}$. An m-dimensional lattice Λ over $\mathbb{Z}$ (resp. R) is a discrete additive subgroup of $\mathbb{Z}$ (resp. R). Equivalently, $\Lambda = \mathcal{L}(\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}) := \{\sum_{i\in[k]} x_i \boldsymbol{b}_i : x_i \in \mathbb{Z}\}$ for a set of linearly independent vectors $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k \in \mathbb{Z}^m$ (resp. R^m), which is referred to as a basis of Λ. The size k is the *rank* of the lattice Λ. We say Λ is a *full rank* lattice if $k = m$ (resp. $k = mN$ for Λ over R). For any $a \in \mathbb{Z}^m$ (resp. R^m), $\Lambda + a$ is a *coset* of Λ. The *dual lattice* of Λ is denoted as $\Lambda^* = \{\boldsymbol{x} \in \mathsf{Span}(\Lambda) : \forall\, \boldsymbol{y} \in \Lambda, \langle \boldsymbol{x}, \boldsymbol{y}\rangle \in \mathbb{Z}\}$. A Λ-subspace is the linear span of some subset of Λ, i.e., a subspace S such that $S = \mathsf{Span}(S \cap \Lambda)$. For any two vectors $\boldsymbol{v} \in \mathbb{Z}^m$ (resp. R^m) and $\boldsymbol{u} \in \mathbb{Z}^n$ (resp. R^n), denote $\boldsymbol{v} \otimes \boldsymbol{u} := (v_1u_1, \ldots, v_1u_n, \ldots, v_mu_1, \ldots, v_mu_n) \in \mathbb{Z}^{mn}$ (resp. R^{mn}). For any two lattices $\Lambda \subseteq \mathbb{Z}^m$ (resp. R^m) and $\Lambda' \subseteq \mathbb{Z}^n$ (resp. R^n), denote their tensor product as $\Lambda \otimes \Lambda'$, which is the smallest lattice over $\mathbb{Z}^{mn}$ (resp. R^{mn}) that contains $\{\boldsymbol{x} \otimes \boldsymbol{y} : \boldsymbol{x} \in \Lambda, \boldsymbol{y} \in \Lambda'\}$.

Further, for a lattice $\Lambda \subseteq R^m$, we say Λ is a R-lattice if and only if Λ is a R-module, or equivalently, $r\boldsymbol{x} \in \Lambda$ for any $r \in R$ and $\boldsymbol{x} \in \Lambda$. For a matrix $A \in R_q^{k\times m}$, we define the R-lattice $\Lambda_q^{\perp}(A) \subseteq R^m$ as

$$\Lambda_q^{\perp}(A) := \{\boldsymbol{x} \in R^m : A\boldsymbol{x} = 0 \bmod q\}\ .$$

We know $\Lambda_q^{\perp}(A)$ has full-rank since $qR^m \subseteq \Lambda_q^{\perp}(A)$.

For a positive definite matrix $\Sigma \in \mathbb{R}^{m\times m}$ (resp. an invertible positive definite matrix $\Sigma \in K_{\mathbb{R}}^{m\times m}$) and a vector $\boldsymbol{c} \in \mathbb{R}^n$ (resp. $K_{\mathbb{R}}^m$), we define the function $\rho_{\Sigma,\boldsymbol{c}}$ over $\mathbb{R}^m$ (resp. $K_{\mathbb{R}}^m$) as

$$\rho_{\Sigma,\boldsymbol{c}}(\boldsymbol{x}) := \exp\left(-\pi \left\langle \boldsymbol{x} - \boldsymbol{c}, \Sigma^{-1}(\boldsymbol{x} - \boldsymbol{c})\right\rangle\right)\ .$$

Then, we denote $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}$ as the discrete Gaussian distribution over a lattice coset $\Lambda + \boldsymbol{a} \subseteq \mathbb{Z}^m$ (resp. R^m) with covariance matrix Σ, centered at $\boldsymbol{c} \in \mathbb{R}^m$, where for $\boldsymbol{x} \in \Lambda + \boldsymbol{a}$, we define

$$\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}(\boldsymbol{x}) := \mathsf{Pr}[\boldsymbol{x} \leftarrow_{\$} \mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}] = \frac{\rho_{\Sigma,\boldsymbol{c}}(\boldsymbol{x})}{\rho_{\Sigma,\boldsymbol{c}}(\Lambda + \boldsymbol{a})}$$

where $\rho_{\Sigma,\boldsymbol{c}}(\Lambda + \boldsymbol{a}) = \sum_{\boldsymbol{x}\in\Lambda+\boldsymbol{a}} \rho_{\Sigma,\boldsymbol{c}}(\boldsymbol{x})$. For $\Lambda + \boldsymbol{a} \subseteq R^m$, we denote $\mathscr{D}^{m,\mathrm{mod}\ q}_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}(\boldsymbol{x})$ as the distribution of $(\boldsymbol{x} \bmod q) \in R_q^m$ for $\boldsymbol{x}$ sampled from $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}$.

The following lemma shows that a discrete Gaussian distribution over $K_{\mathbb{R}}$ can be viewed as a discrete Gaussian distribution over $\mathbb{R}$ via the coefficient embedding ϕ.

Lemma 2. *For a random variable $\boldsymbol{x} \in K_{\mathbb{R}}^m$, the distribution of $\boldsymbol{x}$ is $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}$ for some lattice coset $\Lambda + \boldsymbol{a} \subseteq R^m$, an invertible positive definite matrix $\Sigma \in K_{\mathbb{R}}^{m\times m}$, and vector $\boldsymbol{c} \in K_{\mathbb{R}}^m$ if and only if the distribution of $\phi(\boldsymbol{x})$ is $\mathscr{D}^{mN}_{\phi(\Lambda+\boldsymbol{a}),\phi_{\mathsf{M}}(\Sigma),\phi(\boldsymbol{c})}$.*

Proof. For any $\boldsymbol{v} \in K_{\mathbb{R}}^m$,

$$\begin{aligned}\rho_{\phi_{\mathsf{M}}(\Sigma),\phi(\boldsymbol{c})}(\phi(\boldsymbol{v})) &= \exp\left(-\pi \left\langle \phi(\boldsymbol{v} - \boldsymbol{c}), \phi_{\mathsf{M}}(\Sigma)^{-1}\phi(\boldsymbol{v} - \boldsymbol{c})\right\rangle\right) \\ &= \exp\left(-\pi \left\langle \boldsymbol{v} - \boldsymbol{c}, \Sigma^{-1}(\boldsymbol{v} - \boldsymbol{c})\right\rangle\right) \\ &= \rho_{\Sigma,\boldsymbol{c}}(\boldsymbol{v})\ .\end{aligned}$$

Therefore, for any $\boldsymbol{x} \in \Lambda + \boldsymbol{a}$, $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}(\boldsymbol{x}) = \mathscr{D}^{mN}_{\phi(\Lambda+\boldsymbol{a}),\phi_{\mathsf{M}}(\Sigma),\phi(\boldsymbol{c})}(\phi(\boldsymbol{x}))$. □

Game $\mathrm{MSIS}^{\mathcal{A}}_{q,k,m,\beta}$:
$A \leftarrow\!\$\, R_q^{k\times m}$; $\boldsymbol{x} \leftarrow \mathcal{A}(A)$
Return $(\boldsymbol{x} \neq 0 \;\wedge\; \|\boldsymbol{x}\| \leqslant \beta \;\wedge\; A\boldsymbol{x} = 0)$

Fig. 1. The module-SIS problem.

Also, we make some remarks about the notations we will use throughout the paper. When $\Sigma = \sigma^2 \mathbb{I}_m$ for $\sigma \in \mathbb{R}$, we will use $\rho_{\sigma,\boldsymbol{c}}$ and $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\sigma,\boldsymbol{c}}$ as $\rho_{\Sigma,\boldsymbol{c}}$ and $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}$, respectively. If the center $\boldsymbol{c} = 0$, then we omit the subscript $\boldsymbol{c}$ from $\rho_{\Sigma,\boldsymbol{c}}$ and $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}$. Moreover, when $\Lambda + \boldsymbol{a} = \mathbb{Z}^m$ (resp. $\Lambda + \boldsymbol{a} = R^m$), we omit $\Lambda + \boldsymbol{a}$ from the subscript of $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}$.

The smoothing parameter of a lattice Λ with respect to $\varepsilon > 0$, denoted by $\eta_\varepsilon(\Lambda)$, is the smallest $s > 0$ such that $\rho_{1/s}(\Lambda^* \backslash \{0\}) \leqslant \varepsilon$. Throughout the paper, we set $\varepsilon = 2^{-2\kappa}$.

We borrow the following lemma from [AGHS13] that bounds the ℓ_2-norm of discrete Gaussian random variables and adapt it to lattices over $K_\mathbb{R}$ by Lemma 2.

Lemma 3 (Lemma 3 of [AGHS13] adapted to $K_\mathbb{R}$). *For any $\varepsilon \in (0,1)$, a lattice $\Lambda \subseteq R^m$, $\boldsymbol{c} \in K^m_\mathbb{R}$, and $\sigma \geqslant \eta_\varepsilon(\Lambda)$, then*

$$\mathsf{Pr}[\|\boldsymbol{x} - \boldsymbol{c}\| \geqslant \sigma\sqrt{mN} \;:\; \boldsymbol{x} \leftarrow\!\$\, \mathscr{D}_{\Lambda,\sigma,\boldsymbol{c}}] \leqslant \frac{1+\varepsilon}{1-\varepsilon} \cdot 2^{-mN} \, .$$

We also borrow the following lemmas from [BTT22] and [GPV08] that bound the smoothing parameters.

Lemma 4 (Lemma 2.5 of [BTT22]). *Let q be an odd integer and A a uniformly random matrix in $R_q^{k\times m}$, $k < m$. Then, for any $\varepsilon > 0$, except with probability at most 2^{-N} on the choice of A, we have*

$$\eta_\varepsilon(\Lambda_q^\perp(A)) \leqslant \frac{8}{\sqrt{\pi}} q^{\frac{k}{m}} \sqrt{N \log(2mN(1+1/\varepsilon))} \, .$$

Lemma 5 (Lemma 2.6 of [GPV08]). *For any full-rank lattice Λ in $\mathbb{R}^m$ and $\varepsilon > 0$,*

$$\eta_\varepsilon(\Lambda) \leqslant \frac{\sqrt{\log\left(2m(1+1/\varepsilon)\right)/\pi}}{\lambda_1^\infty(\Lambda^*)} \, ,$$

where $\lambda_1^\infty(\Lambda^*)$ denotes the ℓ_2 norm of the shortest non-zero vector in the ℓ_∞ norm in the dual lattice Λ^*.

2.4 Assumptions

We recall the module short integer solution (MSIS) problem (defined in Figure 1). The advantage of $\mathcal{A}$ for the MSIS problem is defined as

$$\mathsf{Adv}^{\mathrm{msis}}_{q,k,m,\beta}(\mathcal{A}) := \mathsf{Pr}\left[\mathrm{MSIS}^{\mathcal{A}}_{q,k,m,\beta} = 1\right] \, .$$

2.5 Rényi Divergence

We define the notion of Rényi Divergence [Rén61] between two distributions P, Q which we will use in the analysis of the scheme. For a discrete distribution P, we denote the support of P as $\mathrm{Supp}(P) := \{x : P(x) > 0\}$.

Definition 1 (Rényi Divergence). *Let P, Q be two discrete probability distributions such that $\mathrm{Supp}(P) \subseteq \mathrm{Supp}(Q)$ and $\alpha \in [1, +\infty]$. We define the Rényi Divergence of order α, for $\alpha \in (1, \infty)$ as*

$$R_\alpha(P\|Q) := \left(\sum_{x\in Supp(P)} \frac{P(x)^\alpha}{Q(x)^{\alpha-1}}\right)^{\frac{1}{\alpha-1}} .$$

For $\alpha = 1$ and $\alpha = \infty$, we define

$$R_1(P\|Q) := \exp\left(\sum_{x\in\mathrm{Supp}(P)} P(x)\log\frac{P(x)}{Q(x)}\right),\ R_\infty(P\|Q) := \max_{x\in Supp(P)} \frac{P(x)}{Q(x)} .$$

The two following lemmas, from [ASY22] and [Ros20] respectively, give basic properties of the Rényi Divergence.

Lemma 6 (Lemma 2.27 of [ASY22]). *Let $\alpha \in [1, \infty]$ and P, Q be discrete probability distributions with $\mathrm{Supp}(P) \subseteq \mathrm{Supp}(Q)$. Then, the following properties hold:*

- ***Log Positivity:*** $R_\alpha(P\|Q) \geqslant R_\alpha(P\|P) = 1$.
- ***Data Processing Inequality:*** $R_\alpha\left(P^f\|Q^f\right) \leqslant R_\alpha(P\|Q)$ *for any function f, where P^f (and Q^f) denotes the distribution which samples $x \leftarrow\!\!\$\ P$ ($x \leftarrow\!\!\$\ Q$) and outputs $f(x)$.*
- ***Probability Preservation:*** *Let $E \subseteq \mathrm{Supp}(Q)$ be an arbitrary event. Then, for $\alpha \in (1, \infty)$,* $\Pr_{x \leftarrow\$ Q}[E] \geqslant \Pr_{x \leftarrow\$ P}[E]^{\alpha/(\alpha-1)}/R_\alpha(P\|Q)$.
- ***Weak Triangle Inequality:*** *Let P_1, P_2, P_3 be three probability distributions where $\mathrm{Supp}(P_1) \subseteq \mathrm{Supp}(P_2) \subseteq \mathrm{Supp}(P_3)$. Then, we have*

$$R_\alpha(P_1\|P_3) \leqslant \begin{cases} R_\alpha(P_1\|P_2) \cdot R_\infty(P_2\|P_3) \\ R_\infty(P_1\|P_2)^{\frac{\alpha}{\alpha-1}} \cdot R_\alpha(P_2\|P_3) & \text{if } \alpha \in (1,\infty) \end{cases}$$

Lemma 7 (Proposition 2 of [Ros20]). *Let P and Q denote two distributions of a sequence of random variables $(X_1, \ldots, X_n)$. For $1 \leqslant i \leqslant n$, denote $P_{i|x_{[i-1]}}$ (resp. $Q_{i|x_{[i-1]}}$) as the conditional distribution of X_i given $X_{[i-1]} = x_{[i-1]}$. Then, for any $\alpha > 1$,*

$$R_\alpha(P\|Q) \leqslant \prod_{i\in[n]} \max_{x_{[i-1]}} R_\alpha\left(P_{i|x_{[i-1]}}\|Q_{i|x_{[i-1]}}\right) .$$

The following lemma from [TT15] upperbounds the Rényi Divergence between two discrete Gaussian distributions with different centers.

Lemma 8 (Lemma 5 of [TT15]). *For any m-dimensional lattice $\Lambda \subseteq \mathbb{Z}^m$, a positive definite $\Sigma \in \mathbb{R}^{m\times m}$, and two vectors $\mathbf{c}, \mathbf{c}' \in \mathbb{R}^m$, let $P = \mathscr{D}^m_{\Lambda,\Sigma,\mathbf{c}}$ and $Q = \mathscr{D}^m_{\Lambda,\Sigma,\mathbf{c}'}$. If $\mathbf{c}, \mathbf{c}' \in \Lambda$, set $\varepsilon = 0$. Otherwise, fix $\varepsilon \in (0, 1)$ and assume $\sqrt{\sigma_{\min}(\Sigma)} \geqslant \eta_\varepsilon(\Lambda)$ with $\sigma_{\min}(\Sigma) := \inf_{x\in\mathbb{R}^m,\|x\|=1} \|\Sigma x\|$ denoting the* smallest *singular value of Σ. Then,*

$$R_\alpha(P\|Q) \leqslant \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \exp\left(\alpha\pi\frac{\|\mathbf{c}-\mathbf{c}'\|^2}{\sigma_{\min}(\Sigma)}\right) .$$

By Lemma 2, we derive the following lemma, which adapts the above to lattices over rings.

Lemma 9. *For any m-dimensional lattice coset $\Lambda+\boldsymbol{a} \subseteq R^m$ and any integer $q>0$, an invertible positive definite $\Sigma \in K_{\mathbb{R}}^{m\times m}$, and two vectors $\boldsymbol{c},\boldsymbol{c}' \in R_q^m$, let $P = \mathscr{D}_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}}^{m,\bmod q}$ and $Q = \mathscr{D}_{\Lambda+\boldsymbol{a},\Sigma,\boldsymbol{c}'}^{m,\bmod q}$. If $\boldsymbol{c},\boldsymbol{c}' \in \Lambda+\boldsymbol{a}$, set $\varepsilon=0$. Otherwise, fix $\varepsilon \in (0,1)$ and assume $\sqrt{\sigma_{\min}(\Sigma)} \geqslant \eta_\varepsilon(\Lambda)$. Then,*

$$R_\alpha\left(P\|Q\right) \leqslant \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \exp\left(\alpha\pi\frac{\|\boldsymbol{c}-\boldsymbol{c}'\|^2}{\sigma_{\min}(\Sigma)}\right) .$$

Proof. We can w.l.o.g. assume $\boldsymbol{a}=\boldsymbol{0}$, since P (resp. Q) can be viewed as the distribution of $\boldsymbol{x}+\boldsymbol{a}$ for $\boldsymbol{x}$ sampled from $\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{c}-\boldsymbol{a}}^{m,\bmod q}$ (resp. $\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{c}'-\boldsymbol{a}}^{m,\bmod q}$) and thus $R_\alpha\left(P\|Q\right) = R_\alpha\left(\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{c}-\boldsymbol{a}}^{m,\bmod q}\|\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{c}'-\boldsymbol{a}}^{m,\bmod q}\right)$. For any $\boldsymbol{c},\boldsymbol{c}' \in R_q^m$, there exists $\boldsymbol{v},\boldsymbol{v}' \in R^m$ such that $\boldsymbol{c} \equiv \boldsymbol{v} \bmod q$, $\boldsymbol{c}' \equiv \boldsymbol{v}' \bmod q$, and $\|\boldsymbol{c}-\boldsymbol{c}'\| = \|\boldsymbol{v}-\boldsymbol{v}'\|$. Then, we have $P = \mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}}^{m,\bmod q}$ and $Q = \mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}'}^{m,\bmod q}$. Since $\sigma_{\min}(\phi_{\mathsf{M}}(\Sigma)) = \sigma_{\min}(\Sigma)$, by Lemmas 2 and 8,

$$\begin{aligned}
R_\alpha\left(\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}}^{m}\|\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}'}^{m}\right) &= R_\alpha\left(\mathscr{D}_{\phi(\Lambda),\phi_M(\Sigma),\phi(\boldsymbol{v})}^{mN}\|\mathscr{D}_{\phi(\Lambda),\phi_M(\Sigma),\phi(\boldsymbol{v}')}^{mN}\right)\\
&\leqslant \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \cdot \exp\left(\alpha\pi\frac{\|\phi(\boldsymbol{v})-\phi(\boldsymbol{v}')\|^2}{\sigma_{\min}(\phi_M(\Sigma))}\right)\\
&= \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \cdot \exp\left(\alpha\pi\frac{\|\boldsymbol{v}-\boldsymbol{v}'\|^2}{\sigma_{\min}(\Sigma)}\right) .
\end{aligned}$$

Therefore, by data processing inequality from Lemma 6,

$$\begin{aligned}
R_\alpha\left(P\|Q\right) = R_\alpha\left(\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}}^{m,\bmod q}\|\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}'}^{m,\bmod q}\right) &\leqslant R_\alpha\left(\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}}^{m}\|\mathscr{D}_{\Lambda,\Sigma,\boldsymbol{v}'}^{m}\right)\\
&\leqslant \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \exp\left(\alpha\pi\frac{\|\boldsymbol{v}-\boldsymbol{v}'\|^2}{\sigma_{\min}(\Sigma)}\right)\\
&= \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \exp\left(\alpha\pi\frac{\|\boldsymbol{c}-\boldsymbol{c}'\|^2}{\sigma_{\min}(\Sigma)}\right) .
\end{aligned}$$

□

2.6 Linear transformations of discrete Gaussian random variables

We adopt the notation $P \overset{\varepsilon}{\approx} Q$ from [GMPW20]: for any two distributions P, Q with the same support and $\varepsilon > 0$, we say that $P \overset{\varepsilon}{\approx} Q$ if and only if $\max_{x\in\mathrm{Supp}(P)} |\log P(x) - \log Q(x)| \leqslant \log(1+\varepsilon)$, or equivalently, $\max(R_\infty(P\|Q), R_\infty(Q\|P)) \leqslant 1+\varepsilon$. Note that if $P \overset{\varepsilon}{\approx} Q$, then the statistical distance between P and Q is bounded by $\varepsilon/2$, i.e., $\frac{1}{2}\sum_{x\in\mathrm{Supp}(P)} |P(x)-Q(x)| \leqslant \varepsilon/2$. The following lemma shows that the distribution of a linear transformation of discrete Gaussian random variables is still close to a discrete Gaussian distribution. The proof of the lemma, given in Appendix A, follows a similar proof from [GMPW20].

Lemma 10. *For any $\varepsilon \in (0,1)$ defining $\varepsilon' = 2\varepsilon/(1-\varepsilon)$, $\sigma > 0$, lattice coset $\Lambda+\boldsymbol{a} \subseteq K_{\mathbb{R}}^m$, and matrix $T \in K^{k\times m}$ such that $\phi_{\mathsf{M}}(T)$ has full-row-rank,* $\ker(T)$ *is a Λ-subspace and $\eta_\varepsilon(\Lambda \cap \ker(T)) \leqslant \sigma$, we have*

$$T \cdot \mathscr{D}_{\Lambda+\boldsymbol{a},\sigma}^{m} \overset{\varepsilon'}{\approx} \mathscr{D}_{T\Lambda+T\boldsymbol{a},\sigma^2TT^\dagger}^{k} ,$$

where $T \cdot \mathscr{D}^m_{\Lambda+\boldsymbol{a},\sigma}$ *denotes the distribution of* Tx *for* x *sampled from* $\mathscr{D}^m_{\Lambda+\boldsymbol{a},\sigma}$ *and* $T\Lambda := \{Tx | x \in \Lambda\}$ *is a lattice over* $K^m_{\mathbb{R}}$.

We use the above lemma to show the following, where we consider a particular set of discrete Gaussian variables and linear transformations that are later used in our security proof.

Lemma 11. *For any constant* $\varepsilon \in (0,1)$, $\sigma_0 > 0$, *full-rank* R*-lattice* $\Lambda \subseteq R^m$ *with* $\eta_\varepsilon(\Lambda) \leqslant \sigma_0/(2\sqrt{3mN})$, *arbitrary elements* $\boldsymbol{s}_0, \boldsymbol{s}_1, \ldots, \boldsymbol{s}_\ell \in R^m$ *and* $b_1, \bar{b}_1, \ldots, b_\ell, \bar{b}_\ell \in \mathcal{S}_{\mathsf{b}}$ *(defined in Lemma 1) such that* $(b_1, \ldots, b_\ell) \neq (\bar{b}_1, \ldots, \bar{b}_\ell)$, *let* $\boldsymbol{r}_0, \boldsymbol{r}_1, \ldots, \boldsymbol{r}_\ell$ *be independent samples with* $\boldsymbol{r}_i \leftarrow_\$ \mathscr{D}^m_{\Lambda+\boldsymbol{s}_i,\sigma_0}$, *and* $T = \begin{pmatrix} 1 & b_1 & \cdots & b_\ell \\ 1 & \bar{b}_1 & \cdots & \bar{b}_\ell \end{pmatrix}$ *and* $(\boldsymbol{y}, \bar{\boldsymbol{y}}) = (T \otimes \mathbb{I}_m) \cdot (\boldsymbol{r}_0, \ldots, \boldsymbol{r}_\ell) \in R^{2m}$. *Denote the joint distribution of* $(\boldsymbol{y}, \bar{\boldsymbol{y}})$ *as* D. *Then,*

$$D \overset{\varepsilon'}{\approx} \mathscr{D}^{2m}_{(T\otimes\mathbb{I}_m)\Lambda^{\ell+1}+(\boldsymbol{S},\bar{\boldsymbol{S}}),\Sigma\otimes\mathbb{I}_m} ,$$

where $\varepsilon' = \frac{2((1+\varepsilon)^\ell - 1)}{2-(1+\varepsilon)^\ell}$, $\Lambda^{\ell+1} := \{(\boldsymbol{x}_0, \ldots, \boldsymbol{x}_\ell) : \forall\, i \in [0..\ell], \boldsymbol{x}_i \in \Lambda\}$, *which is a* $(\ell+1)m$*-dimensional lattice over* R, $(\boldsymbol{S}, \bar{\boldsymbol{S}}) = (T \otimes \mathbb{I}_m) \cdot (\boldsymbol{s}_0, \ldots, \boldsymbol{s}_\ell)$, *and* $\Sigma = \sigma_0^2 TT^\dagger \in K^{2\times 2}_{\mathbb{R}}$ *is invertible and positive definite.*

Moreover, denote D_1 *as the marginal distribution of* $\boldsymbol{y}$ *and* $D_{2|\boldsymbol{y}_0}$ *as the distribution of* $\bar{\boldsymbol{y}}$ *conditioning on* $\boldsymbol{y} = \boldsymbol{y}_0$ *for any* $\boldsymbol{y}_0 \in \Lambda + \boldsymbol{S}$, *and we have*

$$D_1 \overset{\varepsilon'}{\approx} \mathscr{D}^m_{\Lambda+\boldsymbol{S},\sigma=\sqrt{\Sigma_{11}}} ,\ D_{2|\boldsymbol{y}_0} \overset{\varepsilon'}{\approx} \mathscr{D}^m_{\mathcal{I}\otimes\Lambda+\boldsymbol{y}_0+\bar{\boldsymbol{S}}-\boldsymbol{S}, \frac{\Delta(\Sigma)}{\Sigma_{11}}\cdot\mathbb{I}_m, \frac{\Sigma_{12}}{\Sigma_{11}}\boldsymbol{y}_0} ,$$

where $\mathcal{I} \subseteq R$ *is the ideal generated by* $b_1 - \bar{b}_1, \ldots, b_\ell - \bar{b}_\ell$, Σ_{ij} *denotes the entry in the* i*-th row and* j*-th column of* Σ, $\Delta(\Sigma) = \Sigma_{11}\Sigma_{22} - \Sigma_{12}\Sigma_{21}$ *denotes the determinant of* Σ, *and* $\Sigma_{11} = \Sigma_{22} = \sigma_0^2(1+\ell)$.

Proof. We first show the statement on the distribution D. Since the distribution of $(\boldsymbol{r}_0, \ldots, \boldsymbol{r}_\ell)$ is $\mathscr{D}^{(1+\ell)m}_{\Lambda^{\ell+1}+(\boldsymbol{s}_0,\ldots,\boldsymbol{s}_\ell),\sigma_0}$, by Lemma 10, we just need to show that $\phi_{\mathsf{M}}(T \otimes \mathbb{I}_m)$ has full-row-rank and $\ker(T \otimes \mathbb{I}_m)$ is a $\Lambda^{\ell+1}$-subspace with $\eta_{\varepsilon''}(\Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)) \leqslant \sigma_0$, where $\varepsilon'' = (1+\varepsilon)^\ell - 1$. Since $(b_1, \ldots, b_\ell) \neq (\bar{b}_1, \ldots, \bar{b}_\ell)$, $\phi_{\mathsf{M}}(T \otimes \mathbb{I}_m)$ has full-row-rank. Assume w.l.o.g $b_1 \neq \bar{b}_1$. For $2 \leqslant j \leqslant \ell$, denote $\boldsymbol{v}_{j-1} = (b_j\bar{b}_1 - b_1\bar{b}_j, \bar{b}_j - b_j, 0, \ldots, 0, b_1 - \bar{b}_1, 0, \ldots, 0)$, where the $(j+1)$-th entry of $\boldsymbol{v}_j$ is set to $b_1 - \bar{b}_1$.

Denote $\Lambda_0 = \mathcal{L}(\{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_{\ell-1}\}) \subseteq R^{\ell+1}$. Then, we have $\Lambda_0 \subseteq \ker(T)$. Also, since for any $\boldsymbol{x} \in \Lambda$ and $(t_0, \ldots, t_\ell) \in \Lambda_0$, $(t_0\boldsymbol{x}, \ldots, t_\ell\boldsymbol{x}) \in \Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)$, we have $\Lambda_0 \otimes \Lambda \subseteq \Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)$, which implies $\mathsf{Span}(\Lambda_0 \otimes \Lambda) \subseteq \mathsf{Span}(\Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)) \subseteq \ker(T \otimes \mathbb{I}_m)$. Therefore, we just need to show that the $\mathsf{Span}(\Lambda^{\ell+1}$ and $\ker(T \otimes \mathbb{I}_m)$ are of the same dimension, which implies $\mathsf{Span}(\Lambda_0 \otimes \Lambda) = \mathsf{Span}(\Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)) = \ker(T \otimes \mathbb{I}_m)$.

Since Λ is a full-rank lattice, there exists a basis $\{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{mN}\}$ of Λ. We now show that $\boldsymbol{e}_1 \otimes \boldsymbol{v}_1, \ldots, \boldsymbol{e}_{mN} \otimes \boldsymbol{v}_1, \ldots, \boldsymbol{e}_1 \otimes \boldsymbol{v}_{\ell-1}, \ldots, \boldsymbol{e}_{mN} \otimes \boldsymbol{v}_{\ell-1}$ are linearly independent. Otherwise, suppose there exists $(a_1, \ldots, a_{mN(\ell-1)}) \neq \mathbf{0} \in \mathbb{R}^{N(\ell-1)}$ such that

$$\sum_{i\in[mN],j\in[\ell-1]} a_{(j-1)mN+i}\boldsymbol{e}_i \otimes \boldsymbol{v}_j = \mathbf{0} .$$

Since there exists $j \in [\ell-1]$ such that $(a_{(j-1)mN+1}, \ldots, a_{jmN}) \neq 0$. Since $v_{j',j+1} = 0$ for $j' \neq j$, we have $\sum_{i\in[mN],j'\in[\ell-1]} a_{(j'-1)mN+i}\boldsymbol{e}_i v_{j',j+1} = \sum_{i\in[mN]} a_{(j-1)mN+i}\boldsymbol{e}_i v_{j,j+1} = 0$. Since $v_{j,j+1} = (b_1 -$

$\bar{b}_1) \neq 0 \in R \subseteq K$, we know $(b_1 - \bar{b}_1)$ is invertible and thus $\sum_{i\in[mN]} a_{(j-1)mN+i}\boldsymbol{e}_i = 0$, which contradicts with the fact that $\boldsymbol{e}_1, \ldots, \boldsymbol{e}_{mN}$ are linearly independent.

Therefore, $\mathsf{Span}(\Lambda_0 \otimes \Lambda)$ is a $\mathbb{R}$-subspace of $K_{\mathbb{R}}^{m(\ell+1)}$ with dimension at least $m(\ell-1)N$. Since $(b_1, \ldots, b_\ell) \neq (\bar{b}_1, \ldots, \bar{b}_\ell)$, $\phi_{\mathsf{M}}(T)$ has full row rank and thus $\ker(\phi_{\mathsf{M}}(T))$ is a $(\ell-1)N$-dimensional $\mathbb{R}$-subspace of $\mathbb{R}^{N(\ell+1)}$. Since $\ker(\phi_{\mathsf{M}}(T)) = \phi(\ker(T))$, we know $\ker(T)$ is a $(\ell-1)N$-dimensional $\mathbb{R}$-subspace of $K_{\mathbb{R}}^{\ell+1}$. Therefore, $\ker(T \otimes \mathbb{I}_m)$ is a $mN(\ell-1)$-dimensional $\mathbb{R}$-subspace of $K_{\mathbb{R}}^{m(\ell+1)}$. Since $\mathsf{Span}(\Lambda_0 \otimes \Lambda) \subseteq \ker(T \otimes \mathbb{I}_m)$, $\mathsf{Span}(\Lambda_0 \otimes \Lambda)$ also has dimension $mN(\ell-1)$, which implies $\mathsf{Span}(\Lambda_0 \otimes \Lambda) = \ker(T \otimes \mathbb{I}_m)$. Since $\mathsf{Span}(\Lambda_0 \otimes \Lambda) \subseteq \mathsf{Span}(\Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)) \subseteq \ker(T \otimes \mathbb{I}_m)$, $\mathsf{Span}(\Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)) = \ker(T \otimes \mathbb{I}_m)$, which means $\ker(T \otimes \mathbb{I}_m)$ is a $\Lambda^{\ell+1}$-subspace.

Since $\Lambda_0 \otimes \Lambda \subseteq \Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)$, $\eta_{\varepsilon''}(\Lambda^{\ell+1} \cap \ker(T \otimes \mathbb{I}_m)) \leqslant \eta_{\varepsilon''}(\Lambda_0 \otimes \Lambda)$. Therefore, it is left to show that $\eta_{\varepsilon''}(\Lambda_0 \otimes \Lambda) \leqslant \sigma_0$. Let $\tilde{\boldsymbol{v}}_i := \boldsymbol{v}_i - \sum_{i'\in[i-1]} \frac{\tilde{\boldsymbol{v}}_{i'}^\dagger \boldsymbol{v}_i}{\tilde{\boldsymbol{v}}_{i'}^\dagger \tilde{\boldsymbol{v}}_{i'}} \tilde{\boldsymbol{v}}_{i'} \in K$ for $i \in [\ell-1]$, where in particular $\tilde{\boldsymbol{v}}_1 = \boldsymbol{v}_1$. Here $\tilde{\boldsymbol{v}}_i \neq 0$ since $\boldsymbol{v}_{i,i+1} \neq 0$ and $\boldsymbol{v}_{i',i+1} = 0$ for $i' \in [i-1]$, and $\tilde{\boldsymbol{v}}_{i'}^\dagger \tilde{\boldsymbol{v}}_{i'}$ is invertible since $\tilde{\boldsymbol{v}}_{i'} \in K$. Then, we have $\tilde{\boldsymbol{v}}_i^\dagger \boldsymbol{v}_j = 0$ for any $i > j \in [\ell-1]$. Denote $\boldsymbol{v} \otimes \Lambda := \{\boldsymbol{v} \otimes \boldsymbol{x} : \boldsymbol{x} \in \Lambda\}$. For any two lattices $\Lambda_1, \Lambda_2 \in K_{\mathbb{R}}^m$, denote their direct sum as $\Lambda_1 + \Lambda_2 := \{\boldsymbol{x} + \boldsymbol{y} : \boldsymbol{x} \in \Lambda_1, \boldsymbol{y} \in \Lambda_2\}$. Then, $\Lambda_0 \otimes \Lambda = \boldsymbol{v}_1 \otimes \Lambda + \cdots + \boldsymbol{v}_{\ell-1} \otimes \Lambda$.

We can show that $\tilde{\boldsymbol{v}}_i \otimes \Lambda$ is the projection of $\boldsymbol{v}_i \otimes \Lambda$ orthogonal to $\mathsf{Span}(\boldsymbol{v}_1 \otimes \Lambda + \cdots + \boldsymbol{v}_{i-1} \otimes \Lambda)$. By the construction of $\tilde{\boldsymbol{v}}_i$, there exists $\lambda_1, \ldots, \lambda_{i-1} \in K$ such that $\boldsymbol{v}_i - \tilde{\boldsymbol{v}}_i = \sum_{j\in[i-1]} \lambda_j \boldsymbol{v}_j$. Therefore, for any $\boldsymbol{x} \in \Lambda$, $\boldsymbol{v}_i \otimes \boldsymbol{x} = \tilde{\boldsymbol{v}}_i \otimes \boldsymbol{x} + (\boldsymbol{v}_i - \tilde{\boldsymbol{v}}_i) \otimes \boldsymbol{x}$. Then, $\tilde{\boldsymbol{v}}_i \otimes \boldsymbol{x}$ is orthogonal to $\mathsf{Span}(\boldsymbol{v}_1 \otimes \Lambda + \cdots + \boldsymbol{v}_{i-1} \otimes \Lambda)$, since for any $\boldsymbol{y} \in \Lambda$ and $j \in [i-1]$, $\langle \tilde{\boldsymbol{v}}_i \otimes \boldsymbol{x}, \boldsymbol{v}_j \otimes \boldsymbol{y} \rangle = \sum_{i'\in[\ell+1]} \left\langle \boldsymbol{x}, \tilde{v}_{i,i'}^* \tilde{v}_{j,i'} \boldsymbol{y} \right\rangle = \left\langle \boldsymbol{x}, (\tilde{\boldsymbol{v}}_i^\dagger \tilde{\boldsymbol{v}}_j) \boldsymbol{y} \right\rangle = \langle \boldsymbol{x}, 0 \cdot \boldsymbol{y} \rangle = 0$. Also, $(\boldsymbol{v}_i - \tilde{\boldsymbol{v}}_i) \otimes \boldsymbol{x} = \sum_{j\in[i-1]} \lambda_j \boldsymbol{v}_j \otimes \boldsymbol{x} = \sum_{j\in[i-1]} (\sum_{k\in[0..(N-1)]} \lambda_{j,k} X^k) \boldsymbol{v}_j \otimes \boldsymbol{x} = \sum_{j\in[i-1]} \sum_{k\in[0..(N-1)]} \lambda_{j,k} \boldsymbol{v}_j \otimes (X^k \boldsymbol{x}) \in \mathsf{Span}(\boldsymbol{v}_1 \otimes \Lambda + \cdots + \boldsymbol{v}_{i-1} \otimes \Lambda)$, since Λ is a R-lattice and thus $(X^k \boldsymbol{x}) \in \Lambda$. Therefore, $\tilde{\boldsymbol{v}}_i \otimes \boldsymbol{x}$ is the projection of $\boldsymbol{v}_i \otimes \boldsymbol{x}$ orthogonal to $\mathsf{Span}(\boldsymbol{v}_1 \otimes \Lambda + \cdots + \boldsymbol{v}_{i-1} \otimes \Lambda)$.

Thus, by applying the following lemma from [MP13] $\ell - 1$ times, we have $\eta_{\varepsilon''}(\Lambda_0 \otimes \Lambda) \leqslant \eta_{(1+\varepsilon)^{\ell-1}-1}(\Lambda_0 \otimes \Lambda) \leqslant \max_{i\in[\ell-1]} \eta_\varepsilon(\tilde{\boldsymbol{v}}_j \otimes \Lambda)$.

Lemma 12 (Lemma 2.6 [MP13] adapted to $K_{\mathbb{R}}$). *For any lattices $\Lambda_1, \Lambda_2 \subseteq R^m$, let $\tilde{\Lambda}_2$ be the projection of Λ_2 orthogonal to $\mathsf{Span}(\Lambda_1)$, and we have*

$$\eta_\varepsilon(\Lambda_1 + \Lambda_2) \leqslant \max\{\eta_{\varepsilon_1}(\Lambda_1), \eta_{\varepsilon_2}(\tilde{\Lambda}_2)\} ,$$

where $\varepsilon = (1+\varepsilon_1)(1+\varepsilon_2) - 1$.

Then, we use the following lemma to bound $\eta_\varepsilon(\tilde{\boldsymbol{v}}_i \otimes \Lambda)$.

Lemma 13. *For any real value $\varepsilon > 0$, a R-lattice $\Lambda \subseteq R^m$ and a vector $\boldsymbol{v} \in K^n$, we have*

$$\eta_\varepsilon(\boldsymbol{v} \otimes \Lambda) \leqslant \sqrt{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})} \cdot \eta_\varepsilon(\Lambda) .$$

Proof (of Lemma 13). We first show that the dual lattice $(\boldsymbol{v} \otimes \Lambda)^* = \frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}} \boldsymbol{v} \otimes \Lambda^*$. Since $(\boldsymbol{v} \otimes \Lambda)^* \in \mathsf{Span}(\boldsymbol{v} \otimes \Lambda) \subseteq \boldsymbol{v} \otimes \mathsf{Span}(\Lambda)$, any element in $(\boldsymbol{v} \otimes \Lambda)^*$ can be represented as $\boldsymbol{v} \otimes \boldsymbol{x}$ where $\boldsymbol{x} \in \mathsf{Span}(\Lambda)$. For $\boldsymbol{v} \otimes \boldsymbol{x} \in (\boldsymbol{v} \otimes \Lambda)^*$, we know $\langle \boldsymbol{v} \otimes \boldsymbol{x}, \boldsymbol{v} \otimes \boldsymbol{y} \rangle \in \mathbb{Z}$ for any $\boldsymbol{y} \in \Lambda$. Therefore, $\left\langle (\boldsymbol{v}^\dagger \boldsymbol{v}) \boldsymbol{x}, \boldsymbol{y} \right\rangle = \langle \boldsymbol{v} \otimes \boldsymbol{x}, \boldsymbol{v} \otimes \boldsymbol{y} \rangle \subseteq \mathbb{Z}$, which implies $(\boldsymbol{v}^\dagger \boldsymbol{v}) \boldsymbol{x} \in \Lambda^*$. Thus, $\boldsymbol{v} \otimes \boldsymbol{x} = \frac{\boldsymbol{v}}{\boldsymbol{v}^\dagger \boldsymbol{v}} \otimes (\boldsymbol{v}^\dagger \boldsymbol{v}) \boldsymbol{x} \in \frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}} \boldsymbol{v} \otimes \Lambda^*$, which implies $(\boldsymbol{v} \otimes \Lambda)^* \subseteq \frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}} \boldsymbol{v} \otimes \Lambda^*$. Also, for any $\boldsymbol{x} \in \Lambda^*$ and $\boldsymbol{y} \in \Lambda$, $\left\langle \frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}} \boldsymbol{v} \otimes \boldsymbol{x}, \boldsymbol{v} \otimes \boldsymbol{y} \right\rangle = \langle \boldsymbol{x}, \boldsymbol{y} \rangle \in \mathbb{Z}$, which implies $\frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}} \boldsymbol{v} \otimes \Lambda^* \subseteq (\boldsymbol{v} \otimes \Lambda)^*$.

For any real value $s > 0$ and $\boldsymbol{x} \in \Lambda^*$, we have

$$\begin{aligned}
\rho_{1/s}(\frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \boldsymbol{x}) &= \exp\left(-\pi \left\langle \frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \boldsymbol{x}, \frac{s^2}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \boldsymbol{x} \right\rangle\right) \\
&\leqslant \exp\left(-\pi \frac{1}{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})} \left\langle \frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \boldsymbol{x}, \frac{s^2 \boldsymbol{v}^\dagger \boldsymbol{v}}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \boldsymbol{x} \right\rangle\right) \\
&= \exp\left(-\pi \left\langle \boldsymbol{x}, \frac{s^2}{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})}\boldsymbol{x} \right\rangle\right) \\
&= \rho_{\sqrt{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})})/s}(\boldsymbol{x}) ,
\end{aligned}$$

where the first inequality is due to the fact that $\langle \boldsymbol{y}, (\boldsymbol{v}^\dagger \boldsymbol{v})\boldsymbol{y} \rangle \leqslant \sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v}) \langle \boldsymbol{y}, \boldsymbol{y} \rangle$ for any $\boldsymbol{y} \in K_{\mathbb{R}}^m$. The above implies $\rho_{1/s}(\frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \Lambda^* \backslash \{\boldsymbol{0}\}) \leqslant \rho_{\sqrt{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})})/s}(\Lambda^* \backslash \{\boldsymbol{0}\})$. By letting $s = \sqrt{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})} \cdot \eta_\varepsilon(\Lambda)$, $\eta_\varepsilon(\boldsymbol{v} \otimes \Lambda) = \eta_\varepsilon(\frac{1}{\boldsymbol{v}^\dagger \boldsymbol{v}}\boldsymbol{v} \otimes \Lambda^*) \leqslant s = \sqrt{\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v})}) \cdot \eta_\varepsilon$. □

It holds that $\sigma_{\max}(\boldsymbol{v}^\dagger \boldsymbol{v}) \leqslant \|\boldsymbol{v}\|_1^2$ for $\boldsymbol{v} \in K^m$, since for any $a \in K_{\mathbb{R}}$,

$$\begin{aligned}
\langle a, (\boldsymbol{v}^\dagger \boldsymbol{v})a \rangle = \langle \boldsymbol{v} \cdot a, \boldsymbol{v} \cdot a \rangle = \|\boldsymbol{v} \cdot a\|^2 \leqslant \sum_{i \in [m]} \|v_i a\|^2 &= \sum_{i \in [m]} \left\| \sum_{j \in [N]} (v_{i,j} X^{j-1})a \right\|^2 \\
\leqslant \sum_{i \in [m]} (\sum_{j \in [N]} \|v_{i,j} X^{j-1} a\|)^2 &= \sum_{i \in [m]} (\sum_{j \in [N]} |v_{i,j}| \, \|a\|)^2 \\
\leqslant (\sum_{i \in [m]} \sum_{j \in [N]} |v_{i,j}| \, \|a\|)^2 &= (\|\boldsymbol{v}\|_1 \|a\|)^2 = \|\boldsymbol{v}\|_1^2 \|a\|^2 .
\end{aligned}$$

For any $i \in [\ell - 1]$, since $\|\boldsymbol{v}_i\| \leqslant 2\sqrt{3}$, we have $\|\tilde{\boldsymbol{v}}_i\|_1 \leqslant \sqrt{mN} \|\tilde{\boldsymbol{v}}_i\| \leqslant \sqrt{mN} \|\boldsymbol{v}_i\| \leqslant 2\sqrt{3mN}$. Therefore, $\eta_\varepsilon(\tilde{\boldsymbol{v}}_i \otimes \Lambda) \leqslant \sigma_{\max}(\tilde{\boldsymbol{v}}_i^\dagger \tilde{\boldsymbol{v}}_i) \cdot \eta_\varepsilon \leqslant \|\tilde{\boldsymbol{v}}_i\|_1 (\Lambda) \leqslant \|\boldsymbol{v}_i\| \cdot \eta_\varepsilon(\Lambda) \leqslant 2\sqrt{3mN} \cdot \eta_\varepsilon(\Lambda) \leqslant \sigma_0$, which implies $\eta_{\varepsilon''}(\Lambda_0 \otimes \Lambda) \leqslant \sigma_0$.

For the statement on D_1, denote $T_1 = (1, b_1, \ldots, b_\ell)$, which is the first row of T. Following a similar proof as the first part, we know $D_1 \overset{\varepsilon'}{\approx} \mathscr{D}^m_{(T_1 \otimes \mathbb{I}_m)\Lambda^{\ell+1} + \boldsymbol{S}, \sigma_0^2 T_1 T_1^\dagger \otimes \mathbb{I}_m}$, and we can show the statement since $(T_1 \otimes \mathbb{I}_m)\Lambda^{\ell+1} = \Lambda + b_1\Lambda + \cdots b_\ell \Lambda = \Lambda$ and $\sigma_0^2 T_1 T_1^\dagger = \Sigma_{11} = \sigma_0^2(1 + \sum_{i \in [\ell]} b_i b_i^*) = \sigma_0^2(1 + \ell)$ is a real number.

For $D_{2|\boldsymbol{y}_0}$, we just need to show that assuming $(\boldsymbol{y}, \bar{\boldsymbol{y}})$ is sampled from

$$\mathscr{D}_{(T \otimes \mathbb{I}_m)\Lambda^{\ell+1} + (\boldsymbol{S}, \bar{\boldsymbol{S}}), \Sigma \otimes \mathbb{I}_m} ,$$

the distribution of $\bar{\boldsymbol{y}}$ conditioning on $\boldsymbol{y} = \boldsymbol{y}_0$ is identical to the target distribution for any $\boldsymbol{y}_0 \in \Lambda + \boldsymbol{S}$. It is clear that $\boldsymbol{y}$ is distributed over the lattice coset $C = \{\boldsymbol{x} : (\boldsymbol{y}_0, \boldsymbol{x}) \in (T \otimes \mathbb{I}_m)\Lambda^{\ell+1} + (\boldsymbol{S}, \bar{\boldsymbol{S}})\}$. We first show that $C = \mathcal{I} \otimes \Lambda + \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S}$. For any $\boldsymbol{x} \in C$, there exists $\boldsymbol{z}_0, \ldots, \boldsymbol{z}_\ell \in \Lambda$ such that $\boldsymbol{y}_0 = \boldsymbol{z}_0 + \boldsymbol{S} + \sum_{i \in [\ell]} b_i \boldsymbol{z}_i$ and $\boldsymbol{x} = \boldsymbol{z}_0 + \bar{\boldsymbol{S}} + \sum_{i \in [\ell]} \bar{b}_i \boldsymbol{z}_i$. Therefore, $\boldsymbol{x} = \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S} + \sum_{i \in [\ell]} (\bar{b}_i - b_i)\boldsymbol{z}_i \in \mathcal{I} \otimes \Lambda + \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S}$, which implies $C \subseteq \mathcal{I} \otimes \Lambda + \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S}$. Also, for any $\boldsymbol{x} \in \mathcal{I} \otimes \Lambda + \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S}$, we can represent $\boldsymbol{x} = \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S} + \sum_{i \in [\ell]} (\bar{b}_i - b_i) r_i \boldsymbol{z}_i$ for $r_i \in R$ and $\boldsymbol{z}_i \in \Lambda$. Let $\boldsymbol{z}_0 = \boldsymbol{y}_0 - \boldsymbol{S} - \sum_{i \in [\ell]} b_i r_i \boldsymbol{z}_i \in \Lambda$. Then, $(\boldsymbol{y}_0, \boldsymbol{x}) = (\boldsymbol{z}_0 + \boldsymbol{S} + \sum_{i \in [\ell]} b_i r_i \boldsymbol{z}_i, \boldsymbol{z}_0 + \bar{\boldsymbol{S}} + \sum_{i \in [\ell]} \bar{b}_i r_i \boldsymbol{z}_i) \in C$, which implies that $\mathcal{I} \otimes \Lambda + \boldsymbol{y}_0 + \bar{\boldsymbol{S}} - \boldsymbol{S} \subseteq C$ and thus the two lattice cosets are identical.

It is left to compute the probability of $\bar{\boldsymbol{y}}$ conditioning on $\boldsymbol{y} = \boldsymbol{y}_0$. Since

$$\Sigma^{-1} = \frac{1}{\Delta(\Sigma)} \begin{pmatrix} \Sigma_{22} & -\Sigma_{21} \\ -\Sigma_{12} & \Sigma_{11} \end{pmatrix} ,$$

where $\Delta(\Sigma) = \Sigma_{11}\Sigma_{22} - \Sigma_{12}\Sigma_{21}$, [4] and $\Sigma_{12} = \Sigma_{21}^*$, the probability of $(\boldsymbol{y}, \bar{\boldsymbol{y}})$ is proportional to

$$\begin{aligned} \rho_{\Sigma \otimes \mathbb{I}_m}(\boldsymbol{y}, \bar{\boldsymbol{y}}) &= \exp\left(-\pi \left\langle \begin{pmatrix} \boldsymbol{y} \\ \bar{\boldsymbol{y}} \end{pmatrix}, (\Sigma^{-1} \otimes \mathbb{I}_m) \begin{pmatrix} \boldsymbol{y} \\ \bar{\boldsymbol{y}} \end{pmatrix} \right\rangle\right) \\ &= \exp\left(-\pi \left(\left\langle \boldsymbol{y}, \frac{\Sigma_{22}}{\Delta(\Sigma)} \boldsymbol{y} \right\rangle - \left\langle \bar{\boldsymbol{y}}, \frac{\Sigma_{12}}{\Delta(\Sigma)} \boldsymbol{y} \right\rangle - \left\langle \boldsymbol{y}, \frac{\Sigma_{21}}{\Delta(\Sigma)} \bar{\boldsymbol{y}} \right\rangle + \left\langle \bar{\boldsymbol{y}}, \frac{\Sigma_{11}}{\Delta(\Sigma)} \bar{\boldsymbol{y}} \right\rangle \right)\right) \\ &= \exp\left(-\pi \left(\left\langle \bar{\boldsymbol{y}} - \frac{\Sigma_{12}}{\Sigma_{11}} \boldsymbol{y}, \frac{\Sigma_{11}}{\Delta(\Sigma)} \bar{\boldsymbol{y}} - \frac{\Sigma_{12}}{\Delta(\Sigma)} \boldsymbol{y} \right\rangle \right.\right. \\ &\qquad\qquad \left.\left. + \left\langle \boldsymbol{y}, \frac{\Sigma_{22} - \Sigma_{12}\Sigma_{21}/\Sigma_{11}}{\Delta(\Sigma)} \boldsymbol{y} \right\rangle \right)\right) . \end{aligned}$$

Thus, the probability of $\bar{\boldsymbol{y}}$ conditioning on $\boldsymbol{y} = \boldsymbol{y}_0$ is proportional to

$$\exp\left(-\pi \left\langle \bar{\boldsymbol{y}} - \frac{\Sigma_{21}}{\Sigma_{11}} \boldsymbol{y}_0, \frac{\Sigma_{11}}{\Delta \Sigma} (\bar{\boldsymbol{y}} - \frac{\Sigma_{21}}{\Sigma_{11}} \boldsymbol{y}_0) \right\rangle\right) ,$$

which implies the statement. □

2.7 Other useful lemmas

We also show the following useful lemmas.

Lemma 14. *For any integer $n > 0$ and any $v \in R$ such that $\|v\|_1 \leqslant n$ and $v \neq n$,*

$$\sigma_{\max}((1 + v)^*(1 + v)) \leqslant (n + 1)^2 - 2n/N^2 .$$

Proof. Denote $a = (1 + v)^*(1 + v)$ and $A = \phi_{\mathsf{M}}(a)$, and we have $A_{k,k'} = a_{k'-k}$ for $k' \geqslant k$ and $A_{k,k'} = -a_{N+k'-k}$ for $k' < k$. Since $\langle x, (1 + v)^*(1 + v)x \rangle = \langle \phi(x), A\phi(x) \rangle$ for any $x \in K_{\mathbb{R}}$, $\sigma_{\max}((1 + v)^*(1 + v))$ is equal to the maximum eigenvalue of A.

We find the eigenvalues of A as follows. Denote

$$\boldsymbol{w}_j = (e^{i\pi \frac{(2j-1)}{N}}, \ldots, e^{i\pi \frac{(2j-1)N}{N}}) \in \mathbb{C}^N$$

for $j \in [N - 1]$, where $i := \sqrt{-1}$. Then, $\boldsymbol{w}_j^\dagger \boldsymbol{w}_{j'} = \sum_{k \in [N]} e^{i\pi \frac{2k(j-j')}{N}} = 0$ for $j \neq j' \in [N]$.

[4] Here $\Delta(\Sigma)$ is invertible since $\Sigma = \sigma_0^2 TT^\dagger$ and $\Delta(TT^\dagger) \neq 0 \in K$, which is invertible.

We now show $(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_N)$ are eigenvectors of A. Denote $\boldsymbol{t}_j = A\boldsymbol{w}_j$. Then,

$$\begin{aligned}
t_{j,k} &= \sum_{k' \in [N]} A_{k,k'} w_{j,k'} = -\sum_{k' \in [k-1]} a_{N-k+k'} w_{j,k'} + \sum_{k' \in [k..N]} a_{k'-k} w_{j,k'} \\
&= -\sum_{k' \in [(N-k+1)..(N-1)]} a_{k'} w_{j,k'+k-N} + \sum_{k' \in [0..N-k]} a_{k'} w_{j,k'+k} \\
&= -\sum_{k' \in [(N-k+1)..(N-1)]} a_{k'} e^{i\pi(2j-1)(k'+k-N)/N} + \sum_{k' \in [0..N-k]} a_{k'} e^{i\pi(2j-1)(k'+k)/N} \\
&= \sum_{k' \in [0..(N-1)]} a_{k'} e^{i\pi(2j-1)(k'+k)/N} \\
&= e^{i\pi(2j-1)k/N} \sum_{k' \in [0..(N-1)]} a_{k'} e^{i\pi(2j-1)k'/N} \\
&= w_{j,k} \sum_{k' \in [0..(N-1)]} a_{k'} e^{i\pi(2j-1)k'/N} \ .
\end{aligned}$$

Denote $\lambda_j := \sum_{k' \in [0..(N-1)]} a_{k'} e^{i\pi(2j-1)k'/N}$ and we have $A\boldsymbol{w}_j = \boldsymbol{t}_j = \lambda_j \boldsymbol{w}_j$. Therefore, the eigenvalues of A are $\lambda_1, \ldots, \lambda_N$.

Since $a^* = v^* v = a$, we know $a_k = -a_{N-k}$ for $k \in [1..(N-1)]$. Therefore,

$$\begin{aligned}
\lambda_j &= a_0 + \sum_{k \in [N/2-1]} a_k (e^{i\pi(2j-1)k/N} - e^{i\pi(2j-1)(N-k)/N}) \\
&= a_0 + \sum_{k \in [N/2-1]} a_k 2\cos(\pi(2j-1)k/N) \leqslant a_0 + \sum_{k \in [N/2-1]} a_k 2\cos(\pi/N) \\
&\leqslant a_0 + \sum_{k \in [N/2-1]} a_k 2(1 - 1/N^2) \leqslant a_0 + \sum_{k \in [N-1]} |a_k|(1 - 1/N^2) \\
&= a_0 + (\|a\|_1 - a_0)(1 - 1/N^2) \ .
\end{aligned}$$

Since $a = (1+v)^*(1+v)$, we know $\|a\|_1 \leqslant \|(1+v)\|_1^2 \leqslant (n+1)^2$. If $v = -n$, we know $a = a_0 = (n-1)^2 < n^2 + 1$. Otherwise, since $v \neq n$ and $|v_0| \leqslant \|v\|_1 \leqslant n$, we have $|v_0| \leqslant n - 1$. Therefore, $a_0 = (1+v_0)^2 + \sum_{j=1}^{N-1} |v_j|^2 \leqslant 1 + 2|v_0| + v_0^2 + (\sum_{j=1}^{N-1} |v_j|)^2 \leqslant 1 + 2|v_0| + v_0^2 + (n+1-|v_0|)^2 \leqslant n^2 + 2n + 1 + 2v_0^2 - 2n|v_0| = n^2 + 1 + 2(|v_0| - n)|v_0| \leqslant n^2 + 1$. Therefore, $\lambda_j \leqslant a_0 + (\|a\|_1 - a_0)(1 - 1/N^2) \leqslant n^2 + 1 + 2n(1 - 1/N^2) = (n+1)^2 - 2n/N^2$, which concludes the lemma. □

Lemma 15. *For any $a, b \geqslant 0$ such that $a + b \leqslant 1$ and $\alpha \geqslant 1$, we have $a + b^\alpha \geqslant \frac{1}{\alpha}(a+b)^\alpha$.*

Proof. Let $f(x) = x + b^\alpha$ and $g(x) = \frac{1}{\alpha}(x+b)^\alpha$. Since $f(0) = b^\alpha \geqslant \frac{1}{\alpha} b^\alpha = g(0)$ and $f'(x) = 1 \geqslant (x+b)^{\alpha-1} = g'(x)$ for $x \geqslant 0$, we know $f(x) \geqslant g(x)$ for $0 \leqslant x \leqslant 1 - b$, we have $f(x) \geqslant g(x)$ for $0 \leqslant x \leqslant 1 - b$, which shows the statement. □

3 Linear Secret Sharing Schemes with Small Coefficients

In this section, we first define, in Section 3.1, the notion of linear threshold secret sharing schemes with small coefficients for an abelian group $\mathbb{G}$ (which for our threshold signature $\mathbb{G} = R_q^m$ with its additions as the group operations) and discuss the properties required by our construction. Then, we consider a secret sharing scheme which satisfies the desired properties in Section 3.2 and discuss why other secret sharing schemes do not apply to our case in Section 3.3.

3.1 Definitions

We first give a brief explanation on the notations used in this section. We consider the group $\mathbb{G}$ as a $\mathbb{Z}$-module and adopt the additive notation with 0 as the neutral element. Additionally, for a vector $\boldsymbol{g} \in \mathbb{G}^K$ and a matrix $M \in \mathbb{Z}^{L\times K}$, $M\boldsymbol{g}$ denotes $(\sum_{j=1}^{K} M_{1,j} \cdot g_j, \ldots, \sum_{j=1}^{K} M_{L,j} \cdot g_j)^T \in \mathbb{G}^L$, and for $g \in \mathbb{G}$ and a vector $\boldsymbol{u} \in \mathbb{Z}^K$, $\boldsymbol{u} \cdot g$ denotes $(u_1 \cdot g, \ldots, u_K \cdot g)^T$. Now, we give the following definition for linear threshold secret sharing schemes with small coefficients.

Definition 2 (Linear Threshold Secret Sharing with Small Coefficients). *Let $1 < t \leqslant n, L,$ and K be positive integers and $\mathbb{G}$ be an abelian group. A t-out-of-n linear threshold secret sharing scheme $\mathsf{SecSha}_{t,n}$ for $\mathbb{G}$ consists of two algorithms $(\mathsf{Share}, \mathsf{Recon})$ with the following syntax:*

- $\mathsf{Share}(s \in \mathbb{G}; \boldsymbol{\rho} \in \mathbb{G}^K) \Rightarrow (\mathsf{ss}_j)_{j\in[L]} \in \mathbb{G}^L$*: takes as input a secret $s \in \mathbb{G}$ and a randomness vector $\boldsymbol{\rho} \in \mathbb{G}^K$ (sampled uniformly from $\mathbb{G}^K$), and returns the secret shares $(\mathsf{ss}_j)_{j\in[L]}$. We note that each party $i \in [n]$ has a subset of indices $T_i \subseteq [L]$ such that the share of party i is $(\mathsf{ss}_j)_{j\in T_i}$. We say that the individual share size of party i is $|T_i|$, the total share size is L, and the randomness size is K.*
- $\mathsf{Recon}(U, (\mathsf{ss}_j)_{j\in\bigcup_{i\in U} T_i}) \Rightarrow s \in \mathbb{G}$*: takes as input a set $U \subseteq [n]$ with $|U| \geqslant t$ and the secret shares corresponding to each party in U, and returns the reconstructed secret s.*

We require that $\mathsf{SecSha}_{t,n}$ satisfies the following properties:

- ***Linearity**: The sharing algorithm Share can be written as an integer matrix $M \in \mathbb{Z}^{L\times(K+1)}$ mapping a vector $\boldsymbol{v} = (s, \rho_1, \ldots, \rho_K)^T \in \mathbb{G}^{K+1}$ to $M\boldsymbol{v} \in \mathbb{G}^L$. Moreover, for any $U \subseteq [n]$ denote M_U as the matrix M restricted to the rows indexed with $\bigcup_{i\in U} T_i$, the following is also true:*
  - *For any $U \subseteq [n], |U| \geqslant t$, there exists a **reconstruction coefficient** vector $\boldsymbol{\lambda}^U \in \mathbb{Z}^L$ such that $\lambda_j^U = 0$ for $j \notin \bigcup_{i\in U} T_i$ and $(\boldsymbol{\lambda}^U)^T M = (1, 0, \ldots, 0)$. Then, the output of $\mathsf{Recon}(U, \cdot)$ on input $(\mathsf{ss}_j)_{j\in\bigcup_{i\in U} T_i}$ can be written as $\sum_{i\in U} \sum_{j\in T_i} \lambda_j^U \mathsf{ss}_j$. Hence, for $(\mathsf{ss}_j)_{j\in[L]} \leftarrow \mathsf{Share}(s; \boldsymbol{\rho})$ for any $s \in \mathbb{G}$ and $\boldsymbol{\rho} \in \mathbb{G}^K$, we have that $\sum_{i\in U} \sum_{j\in T_i} \lambda_j^U \mathsf{ss}_j = s$.*
  - *For any $U \subseteq [n]$ with $|U| < t$, there exists a vector $\boldsymbol{u} \in \mathbb{Z}^{K+1}$ such that $u_1 = 1$ and $M_U\boldsymbol{u} = \boldsymbol{0}$. We call such $\boldsymbol{u}$ the **sweeping vector** of M_U.*
- ***Small Coefficients**: For the sharing matrix M, its entries are bounded by β_M and the number of non-zero entries in each row is bounded by β_{row}. For any $U \subseteq [n]$ and $|U| \geqslant t$, the reconstruction coefficient vector λ^U has $\|\lambda^U\|_\infty \leqslant \beta_\lambda$. For any $U \subseteq [n]$ and $|U| < t$, there exists a sweeping vector $\boldsymbol{u}$ of M_U such that $\|\boldsymbol{u}\|_\infty \leqslant \beta_u$.*

We point out that our definition differs from prior works in that we did not explicitly define correctness and privacy properties (since we will not use them in the proofs of our construction), and instead give two properties: linearity and small coefficients. The linearity property already implies correctness and privacy, as shown in prior works [KW93, Bei96, CF02] which showed relations between linear secret sharing schemes and span programs. In particular, the first bullet point of linearity implies correctness, while the second bullet point implies privacy.

The small coefficients property is required by the following lemma, which establishes a crucial property used in the security proof of our threshold signature. Notably, fixing two secret keys

$\mathsf{sk}, \mathsf{sk}' \in R_q^m$ with bounded norms and a corrupted subset $U \subseteq [n]$ with $|U| < t$, one can construct a bijection $\Phi_{\mathsf{sk},\mathsf{sk}',U}$ between the set of the randomness used to generate the secret shares of sk and sk' such that: the secret shares given to the corrupted parties is unchanged (item (1)), and the distance between the reconstructed shares for any party is bounded (item (2)).

Lemma 16. *Let* $(\mathsf{Share}, \mathsf{Recon})$ *be a t-out-of-n linear threshold secret sharing with small coefficients for* $\mathbb{G} = R_q^m$. *In particular, let* $M \in \mathbb{Z}^{L \times (K+1)}$ *be the sharing matrix, and* $\beta_M, \beta_{\mathsf{row}}, \beta_\lambda, \beta_u$ *be the bounds for the small coefficients property. Fix any* $U \subseteq [n]$ *with* $|U| < t$, *a matrix* $A \in R_q^{k \times m}$ *and any* $\mathsf{sk}, \mathsf{sk}' \in R_q^m$ *such that* $A\mathsf{sk} = A\mathsf{sk}'$ *and* $\|\mathsf{sk}\|_\infty, \|\mathsf{sk}'\|_\infty \leqslant \beta_{\mathsf{sk}}$. *Then, there exists a bijection* $\Phi_{\mathsf{sk},\mathsf{sk}',U} : (R_q^m)^K \to (R_q^m)^K$, *such that for any* $\boldsymbol{\rho} \in (R_q^m)^K$ *and* $\boldsymbol{\rho}' = \Phi_{\mathsf{sk},\mathsf{sk}',U}(\boldsymbol{\rho})$, *the secret shares* $(\mathsf{ss}_j)_{j\in[L]} \leftarrow \mathsf{Share}(\mathsf{sk}; \boldsymbol{\rho})$ *and* $(\mathsf{ss}'_j)_{j\in[L]} \leftarrow \mathsf{Share}(\mathsf{sk}'; \boldsymbol{\rho}')$ *satisfy:*

(1) $(\mathsf{ss}_j)_{j \in \bigcup_{i\in U} T_i} = (\mathsf{ss}'_j)_{j \in \bigcup_{i\in U} T_i}$

(2) For any $S \subseteq [n]$ *with* $|S| \geqslant t$, *let* $\boldsymbol{\lambda}^S \in \mathbb{Z}^L$ *be the reconstruction coefficients for* $\mathsf{Recon}(S, \cdot)$. *Also, for* $i \in S$, *define* $\boldsymbol{v}_i = \sum_{j\in T_i} \lambda_j^S \mathsf{ss}_j$ *and* $\boldsymbol{v}'_i = \sum_{j\in T_i} \lambda_j^S \mathsf{ss}'_j$, *we have that* $A\boldsymbol{v}_i = A\boldsymbol{v}'_i$, *and*

$$\left\|\boldsymbol{v}_i - \boldsymbol{v}'_i\right\|_\infty \leqslant \beta_{\mathsf{ss}}\beta_{\mathsf{sk}} ,$$

where $\beta_{\mathsf{ss}} = 2|T_i|\beta_M\beta_{\mathsf{row}}\beta_u\beta_\lambda$.

Proof. Let $\boldsymbol{u} \in \mathbb{Z}^{K+1}$ be a sweeping vector for M_U such that $\|\boldsymbol{u}\|_\infty \leqslant \beta_u$ which exists due to our secret sharing definition. Consider the map $\Phi_{\mathsf{sk},\mathsf{sk}',U}$ defined as $\Phi_{\mathsf{sk},\mathsf{sk}',U}(\boldsymbol{\rho}) = \boldsymbol{\rho} + (u_2, \ldots, u_{K+1})^T \cdot (\mathsf{sk}' - \mathsf{sk})$, which we can see is a bijection on $(R_q^m)^K$ as it only shifts $\boldsymbol{\rho}$ by some fixed amount. Now, fix a $\boldsymbol{\rho} \in (R_q^m)^K$ and $\boldsymbol{\rho}' = \Phi_{\mathsf{sk},\mathsf{sk}',U}(\boldsymbol{\rho})$. Consider the secret shares generated using these two randomness. For any $j \in [L]$, denote M_j as the j-th row of M, then

$$\begin{aligned}\mathsf{ss}'_j - \mathsf{ss}_j &= M_j(\mathsf{sk}', \boldsymbol{\rho}'^T)^T - M_j(\mathsf{sk}, \boldsymbol{\rho}^T)^T \\ &= M_j(\mathsf{sk}' - \mathsf{sk}, (u_2, \ldots, u_{K+1})^T \cdot (\mathsf{sk}' - \mathsf{sk}))^T = (M_j\boldsymbol{u}) \cdot (\mathsf{sk}' - \mathsf{sk}) .\end{aligned}$$

Then, since $M_U\boldsymbol{u} = \mathbf{0}$, we have that (1) is true, since

$$(\mathsf{ss}'_j)_{j \in \bigcup_{i\in U} T_i} = (\mathsf{ss}_j)_{j \in \bigcup_{i\in U} T_i} + (M_U\boldsymbol{u}) \cdot (\mathsf{sk}' - \mathsf{sk}) = (\mathsf{ss}_j)_{j \in \bigcup_{i\in U} T_i} .$$

To show (2), for $i \in [n]$, consider $\boldsymbol{v}_i$ and $\boldsymbol{v}'_i$ as defined in the lemma statement. Then,

$$\boldsymbol{v}'_i - \boldsymbol{v}_i = \sum_{j\in T_i} \lambda_j^S(\mathsf{ss}'_j - \mathsf{ss}_j) = \sum_{j\in T_i} \lambda_j^S(M_j\boldsymbol{u}) \cdot (\mathsf{sk}' - \mathsf{sk}) .$$

Since $\sum_{j\in T_i} \lambda_j^S(M_j\boldsymbol{u}) \in \mathbb{Z}$, we have that $A\boldsymbol{v}'_i - A\boldsymbol{v}_i = \left(\sum_{j\in T_i} \lambda_j^S(M_j\boldsymbol{u})\right) \cdot (A\mathsf{sk}' - A\mathsf{sk}) = \mathbf{0} \in R_q^k$, so $A\boldsymbol{v}'_i = A\boldsymbol{v}_i$. Moreover, with $\beta_{\mathsf{ss}} = 2|T_i|\beta_M\beta_{\mathsf{row}}\beta_u\beta_\lambda$, $\|\boldsymbol{v}_i - \boldsymbol{v}'_i\|_\infty \leqslant \left\|\sum_{j\in T_i} \lambda_j^S(M_j\boldsymbol{u})(\mathsf{sk}' - \mathsf{sk})\right\|_\infty \leqslant \beta_{\mathsf{ss}}\beta_{\mathsf{sk}}$. □

3.2 Instantiation

One secret sharing scheme satisfying Definition 2 is the generic construction from Benaloh and Leichter [BL90] which derives a linear secret sharing scheme for any monotone access structure (i.e., for any set S of parties that can recover the secret, any set that contains S can also recover the secret) from a monotone Boolean formula (i.e., a Boolean circuit with only AND and OR gates of fan-in 2 and fan-out 1, but the input wires may have multiple fan-out) f computing such access structure. Damgård and Thorbek [DT06] showed that Benaloh-Leichter secret sharing satisfies the following properties:

(1) Both the number of randomness K and total share size L are at most the size of the formula f.
(2) The sharing matrix M has binary entries, and the number of 1's in each row is at most the depth of f.
(3) The reconstruction coefficients are in $\{-1, 0, 1\}$.
(4) For any $U \subseteq [n]$ with $|U| < t$, the sweeping vector $\boldsymbol{u}$ of M_U has entries in $\{-1, 0, 1\}$.

Regarding the formula computing threshold access structure, a seminal work by Valiant [Val84] gave a probabilistic construction of a monotone formula for majority function ($(n/2, n)$-threshold function) of size $O(n^{5.3})$ and depth $5.3 \log n + O(1)$. Then, Boppana [Bop85] generalized this result to a monotone formula for (t, n)-threshold function of size $O(t'^{4.3} n \log n)$ and depth $\log n + 4.3 \log t' + \log\log n + O(1)$ where $t' = \min(t, n-t)$. Hoory, Magen, and Pitassi [HMP06] improved this to a monotone *circuit* of size $O(t'^2 n \log n)$ and depth $O(\log n)$. However, as pointed out in [BS23], this construction is not a formula (namely, the gates in this circuit have multiple fan-out), so it does not imply a linear secret sharing scheme. Also, it is worth noting that these are probabilistic constructions with success probability 1/2 of realizing the threshold functions. Still, for small n (e.g., $n = 5, 32$ as we consider in this work), we can exhaustively check if a constructed formula correctly computes the threshold function on all inputs.

The following lemma then formalizes the existence of a secret sharing scheme constructed by applying Benaloh and Leichter's construction to Boppana's monotone formula for threshold function.

Lemma 17. *There exists a t-out-of-n linear threshold secret sharing with small coefficients with total share size $L = O(t'^{4.3} n \log n)$ making the individual share size $|T_i| \leqslant O(t'^{4.3} n \log n)$ for $t' = \min(t, n-t)$ and the small coefficient bounds*

$$\beta_M = \beta_\lambda = \beta_u = 1 \text{ and } \beta_{\mathsf{row}} = \log n + 4.3 \log t' + \log\log n + O(1) ,$$

which result in the bound β_{ss} from Lemma 16 of $\beta_{\mathsf{ss}} = O(t'^{4.3} n (\log n)^2)$.

3.3 Discussion on other secret sharing schemes

In this section, we discuss whether other linear secret sharing schemes, such as a recent ramp/near-threshold secret sharing scheme [ANP23] and the tree secret sharing scheme [CCK23], apply to our case.

Applebaum, Nir, and Pinkas [ANP23] recently proposed a ramp/near-threshold black-box secret sharing scheme where a set of at least $t_c n$ parties is guaranteed to recover a secret, while privacy is guaranteed for any set of less than $t_p n$ corrupted parties with $0 < t_p < t_c < 1$. Their secret sharing scheme has the sharing matrix M of the form

$$M = \begin{pmatrix} 0^{L-1} & G \\ 1 & \boldsymbol{a}^T \end{pmatrix} \in \mathbb{Z}^{L \times (K+1)}$$

where $L, K = O(n)$, $G \in \mathbb{Z}^{(L-1) \times K}$ is a matrix with small entries, and each entry of $\boldsymbol{a} \in \mathbb{Z}^K$ is bounded by some constant c. We also remark that the secret share corresponding to the last row of M is public in their scheme. Their reconstruction can be modeled as a $O(n)$-size $O(\log n)$-depth addition circuit, translating to a bound of $\mathsf{poly}(n)$ on the reconstruction coefficients.

For the sweeping vector, fixing a subset $U \subseteq [n]$ where $|U| < t_p n$ and letting M_U and G_U denote the rows of the matrices M and G of which the shares are known to U, they showed that there exists a vector $\boldsymbol{u'} \in \mathbb{Z}^K$ with each entry bounded by some constant b where $G_U \boldsymbol{u'} = \boldsymbol{0}$ and $v = \boldsymbol{a}^T \cdot \boldsymbol{u'} \neq 0 \mod q$ for any prime $q > 2bcK$ (see Claim 4.1 of [ANP23]). This gives us a vector $(v, -\boldsymbol{u'}^T)^T$ with $|v| \leqslant bcK$ such that $M_U(v, -\boldsymbol{u'}^T)^T = 0$. However, since v is not necessarily 1, we only get a sweeping vector $\boldsymbol{u} = v^{-1}(v, -\boldsymbol{u'}^T)^T \mod q$ of which the entries are not guaranteed to be bounded, because v^{-1} can be large in $\mathbb{Z}_q$. To account for division by any v, one can scale up the secret by the least common multiple of $(1, \ldots, bcK)$, but this scaling is estimated to be $2^{O(n)}$.[5]

Tree secret sharing scheme is proposed by Cheon, Cho, and Kim [CCK23] in the context of improving universal thresholdizer. They constructed a $(n+1)/2$-out-of-n linear secret sharing by repeatedly applying η-out-of-$(2\eta - 1)$ Shamir's secret sharing, for any integer $\eta \geqslant 2$, in a tree structure. The tree is of depth $d \geqslant \log_{c_\eta} n + \log_\eta n + O(1)$ with $c_\eta = \frac{2\eta-2}{2^{2\eta-2}} \cdot \binom{2\eta-2}{s-1}$, and the total share size is $O(n^{\log_{c_\eta}(2\eta-1)+\log_\eta(2\eta-1)})$. They showed that their reconstruction coefficients are bounded by $((2\eta-1)!)^{2d}$, amounting to $O(n^{2(\log_{3/2} 6 + \log_2 6)}) \approx O(n^{14})$ for $\eta = 2$. This already exceeds β_{ss} from Lemma 17, so we do not consider their secret sharing as an instantiation.

4 Threshold Signatures

In this section, we first give formal syntax and security definitions for threshold signatures, then present our construction and the security analysis, and finally discuss the concrete parameters and efficiency.

4.1 Syntax and security

We use the formalization proposed by Bellare et al. [BCK+22], which is also used in [TZ23].

<u>Syntax.</u> A (partially) non-interactive threshold signature schemes for n signers and threshold t is a tuple of efficient (randomized) algorithms $\mathsf{TS} = (\mathsf{Setup}, \mathsf{KeyGen}, \mathsf{SPP}, \mathsf{LPP}, \mathsf{LR}, \mathsf{PS}, \mathsf{Agg}, \mathsf{Vf})$ that behave as follows. Signers involved are a leader and n signers. In real-world scenarios, the leader can be one of the signers. The setup algorithm $\mathsf{Setup}(1^\kappa)$ initializes the state st_i for each signer $i \in [n]$ and st_0 for the leader and returns a system parameter par. We assume par is given to all other algorithms implicitly. The key generation algorithm $\mathsf{KeyGen}()$ returns a public verification key pk, and a secret key sk_i for each signer i.

The signing protocol consists of two rounds: a message-independent offline round and an online signing round. In the offline round, any signer i can run $\mathsf{SPP}(\mathsf{st}_i)$ to generate a pre-processing token pp, which is sent to the leader, and the leader runs $\mathsf{LPP}(i, pp, \mathsf{st}_0)$ to update its state st_0 to incorporate token pp. In the online round, for any signer set $SS \subseteq [n]$ with size t and message $\mu \in \{0,1\}^*$, the leader runs $\mathsf{LR}(\mu, SS, \mathsf{st}_0)$ to generate a leader request lr with $lr.\mathsf{msg} = \mu$ and $lr.\mathsf{SS} = SS$ and sends lr to each signer $i \in SS$. Then, each signer i runs $\mathsf{PS}(lr, i, \mathsf{st}_i)$ to generate its partial signature $psig_i$. Finally, the leader computes a signature sig for μ by running $\mathsf{Agg}(\{psig_i\}_{i \in SS})$. The (deterministic) verification algorithm $\mathsf{Vf}(\mathsf{pk}, \mu, sig)$ outputs a bit that indicates whether sig is valid for (pk, μ).

[5] The natural logarithm of $LCM(1, \ldots, x)$ is the second Chebyshev's function which is bounded by $1.03883x$ [RS62].

Game $\text{TS-COR}_{\mathsf{TS}}^{\mathcal{A}}(\kappa)$:

$par \leftarrow \mathsf{Setup}(1^\kappa)$; $(\mathsf{pk}, \{\mathsf{sk}_i\}_{i\in[n]}) \leftarrow \mathsf{KeyGen}()$
For $i \in [n]$ do $\mathsf{st}_i.\mathsf{sk} \leftarrow \mathsf{sk}_i$; $\mathsf{st}_i.\mathsf{pk} \leftarrow \mathsf{pk}$
$(\mu, SS) \leftarrow \mathcal{A}(par, \mathsf{pk}, \{\mathsf{sk}_i\}_{i\in[n]})$
If $SS \nsubseteq [n]$ or $|SS| < t$ then return 0
For $i \in SS$ do $(pp_i, \mathsf{st}_i) \leftarrow \mathsf{SPP}(\mathsf{st}_i)$; $\mathsf{st}_0 \leftarrow \mathsf{LPP}(i, pp_i, \mathsf{st}_0)$
$(lr, \mathsf{st}_0) \leftarrow \mathsf{LR}(\mu, SS, \mathsf{st}_0)$
For $i \in SS$ do $(psig_i, \mathsf{st}_i) \leftarrow \mathsf{PS}(lr, i, \mathsf{st}_i)$
$sig \leftarrow \mathsf{Agg}(\{psig_i\}_{i\in SS})$
Return $\mathsf{Vf}(\mathsf{pk}, \mu, sig) = 0$

Fig. 2. The TS-COR game for a threshold signature scheme TS with threshold t.

Game $\text{TS-UF-0}_{\mathsf{TS}}^{\mathcal{A}}(\kappa)$:

$par \leftarrow \mathsf{Setup}(1^\kappa)$; $\mathrm{H} \leftarrow\!\!\$\ \mathsf{TS.HF}$; $\mathrm{S} \leftarrow \varnothing$
$(\mu, sig) \leftarrow \mathcal{A}^{\textsc{Init},\text{PPO},\text{PSignO},\text{RO}}(par)$
Return $(\mu \notin \mathrm{S} \ \wedge\ \mathsf{Vf}(\mathsf{pk}, \mu, sig) = 1)$

Oracle $\textsc{Init}(CS)$:

Require: $CS \subseteq [n]$ and $|CS| < t$
$HS \leftarrow [n] \backslash CS$
$(\mathsf{pk}, \mathsf{sk}_1, \ldots, \mathsf{sk}_n) \leftarrow \mathsf{KeyGen}()$
For $i \in HS$ do $\mathsf{st}_i.\mathsf{sk} \leftarrow \mathsf{sk}_i$; $\mathsf{st}_i.\mathsf{pk} \leftarrow \mathsf{pk}$
Return $(\mathsf{pk}, \{\mathsf{sk}_i\}_{i\in CS})$

Oracle $\text{PPO}(i)$:

Require: $i \in HS$
$(pp, \mathsf{st}_i) \leftarrow\!\!\$\ \mathsf{SPP}(\mathsf{st}_i)$; $\mathrm{PP}_i \leftarrow \mathrm{PP}_i \cup \{pp\}$
Return pp

Oracle $\text{PSignO}(i, lr)$:

Require: $lr.\mathsf{SS} \subseteq [n]$ and $i \in HS$
$\mu \leftarrow lr.\mathsf{msg}$; $\mathrm{S} \leftarrow \mathrm{S} \cup \{\mu\}$
$(psig, \mathsf{st}_i') \leftarrow\!\!\$\ \mathsf{PS}(lr, i, \mathsf{st}_i)$
Return $psig$

Oracle $\text{RO}(x)$:

Return $\mathrm{H}(x)$

Fig. 3. The TS-UF-0 game for a threshold signature scheme TS.

In summary, an honest execution of the signing protocol between signers in SS and the leader to sign a message $\mu \in \{0,1\}^*$ is represented in the game TS-COR (defined in Figure 2), and we say that TS is *correct* with correctness error δ if for any adversary $\mathcal{A}$ for the game TS-COR, we have $\mathsf{Pr}[\text{TS-COR}_{\mathsf{TS}}^{\mathcal{A}}(\kappa) = 1] \leqslant \delta$.

<u>Security.</u> A hierarchy for security notions of threshold signatures is proposed in [BCK+22]. In this paper, we consider TS-UF-0, which guarantees that an adversary can generate a valid signature sig for μ only if it receives partial signatures from at least one honest signer for μ. We also note that the same security notion is also used in all the prior lattice-based works, such as [GKS24, PKM+24]. Formally, the TS-UF-0 game is defined in Figure 3, where $\mathsf{TS.HF}$ denotes the space of the hash functions used in TS from which the random oracle is drawn. The advantage of $\mathcal{A}$ for the TS-UF-0 game is defined as $\mathsf{Adv}_{\mathsf{TS}}^{\text{ts-uf-0}}(\mathcal{A}, \kappa) := \mathsf{Pr}\left[\text{TS-UF-0}_{\mathsf{TS}}^{\mathcal{A}}(\kappa) = 1\right]$.

4.2 Construction

Our threshold signature scheme $\mathsf{TSL}[\mathsf{SecSha}]$ is shown in Figure 4, where SecSha is a linear secret sharing scheme with small coefficients (see Definition 2), which can be instantiated from Benaloh and Leichter's secret sharing scheme as discussed in Section 3.2. Each T_i and $\lambda_j^{lr.SS}$ are defined by the

$\underline{\mathsf{Setup}(1^\kappa):}$

$A \leftarrow_\$ R_q^{k\times m}$; $par \leftarrow A$
For $i \in [n]$ do
  $\mathsf{st}_0.\mathrm{curPP}_i \leftarrow \varnothing$
  $\mathsf{st}_i.\mathrm{mapPP} \leftarrow ()$
Return par

$\underline{\mathsf{KeyGen}():}$

$\mathsf{sk} \leftarrow_\$ \mathscr{B}^m_{\beta_{\mathsf{sk}}}$; $\mathsf{pk} \leftarrow A\mathsf{sk} \bmod q$
$\{\mathsf{ss}_j\}_{j\in[L]} \leftarrow_\$ \mathsf{SecSha.Share}(\mathsf{sk})$
For $i \in [n]$ do $\mathsf{sk}_i \leftarrow \{\mathsf{ss}_j\}_{j\in T_i}$
Return $(\mathsf{pk}, \{\mathsf{sk}_i\}_{i\in[n]})$

$\underline{\mathsf{SPP}(\mathsf{st}_i):}$

For $j \in [0..\ell]$ do $\boldsymbol{r}_j \leftarrow_\$ \mathscr{D}^m_{\sigma_{\mathsf{r}}}$
For $j \in [0..\ell]$ do $\boldsymbol{R}_j \leftarrow A\boldsymbol{r}_j \bmod q$
$pp \leftarrow \{\boldsymbol{R}_j\}_{j\in[0..\ell]}$
$\mathsf{st}_i.\mathrm{mapPP}(pp) \leftarrow \{\boldsymbol{r}_j\}_{j\in[0..\ell]}$
Return (pp, st_i)

$\underline{\mathsf{LPP}(i, pp, \mathsf{st}_0):}$

$\mathsf{st}_0.\mathrm{curPP}_i \leftarrow \mathsf{st}_0.\mathrm{curPP}_i \cup \{pp\}$
Return st_0

$\underline{\mathsf{LR}(\mu, SS, \mathsf{st}_0):}$

If $\exists\, i \in SS : \mathsf{st}_0.\mathrm{curPP}_i = \varnothing$ then
  Return $\bot$
$lr.\mathsf{msg} \leftarrow \mu$; $lr.\mathsf{SS} \leftarrow SS$
For $i \in SS$ do
  Pick pp_i from $\mathsf{st}_0.\mathrm{curPP}_i$
  $lr.\mathsf{PP}(i) \leftarrow pp_i$
  $\mathsf{st}_0.\mathrm{curPP}_i \leftarrow \mathsf{st}_0.\mathrm{curPP}_i \backslash \{pp_i\}$
Return (lr, st_0)

$\underline{\mathsf{CompPar}(\mathsf{pk}, lr):}$

$\mu \leftarrow lr.\mathsf{msg}$
For $i \in lr.\mathsf{SS}$ do
  $\{b_j\}_{j\in[\ell]} \leftarrow \mathrm{H}_1(\mathsf{pk}, lr)$
  $\{\boldsymbol{R}_{i,j}\}_{j\in[0..\ell]} \leftarrow lr.\mathsf{PP}(i)$
$\boldsymbol{R} \leftarrow \sum_{i\in lr.\mathsf{SS}} \left(\boldsymbol{R}_{i,0} + \sum_{j\in[\ell]} b_j \boldsymbol{R}_{i,j}\right)$
$c \leftarrow \mathrm{H}_2(\mathsf{pk}, \mu, \boldsymbol{R})$
Return $(\boldsymbol{R}, c, \{b_j\}_{j\in[\ell]})$

$\underline{\mathsf{PS}(lr, i, \mathsf{st}_i):}$

$pp_i \leftarrow lr.\mathsf{PP}(i)$
If $\mathsf{st}_i.\mathrm{mapPP}(pp_i) = \bot$ then return $(\bot, \mathsf{st}_i)$
$\{\boldsymbol{r}_j\}_{j\in[0..\ell]} \leftarrow \mathsf{st}_i.\mathrm{mapPP}(pp_i)$
$\mathsf{st}_i.\mathrm{mapPP}(pp_i) \leftarrow \bot$
$(\boldsymbol{R}, c, \{b_j\}_{j\in[\ell]})) \leftarrow \mathsf{CompPar}(\mathsf{st}_i.\mathsf{pk}, lr)$
$\{\mathsf{ss}_j\}_{j\in T_i} \leftarrow \mathsf{st}_i.\mathsf{sk}$
$\boldsymbol{z} \leftarrow \boldsymbol{r}_0 + \sum_{j\in[\ell]} b_j \cdot \boldsymbol{r}_j$
  $+2c \cdot \sum_{j\in T_i} \lambda_j^{lr.\mathsf{SS}} \mathsf{ss}_j \bmod q$
Return $((\boldsymbol{R}, \boldsymbol{z}), \mathsf{st}_i)$

$\underline{\mathsf{Agg}(\mathrm{PS}, \mathsf{st}_0):}$

$\boldsymbol{R} \leftarrow \bot$; $\boldsymbol{z} \leftarrow 0$
For $(\boldsymbol{R}', \boldsymbol{z}') \in \mathrm{PS}$ do
  If $\boldsymbol{R} = \bot$ then $\boldsymbol{R} \leftarrow \boldsymbol{R}'$
  If $\boldsymbol{R} \neq \boldsymbol{R}'$ then return $(\bot, \mathsf{st}_0)$
  $\boldsymbol{z} \leftarrow \boldsymbol{z} + \boldsymbol{z}'$
Return $((\boldsymbol{R}, \boldsymbol{z}), \mathsf{st}_0)$

$\underline{\mathsf{Vf}(\mathsf{pk}, \mu, sig):}$

$(\boldsymbol{R}, \boldsymbol{z}) \leftarrow sig$
If $\|\boldsymbol{z}\| > \beta_{\mathsf{z}}$ then return 0
$c \leftarrow \mathrm{H}_2(\mathsf{pk}, \mu, \boldsymbol{R})$
Return $(A\boldsymbol{z} = \boldsymbol{R} + 2c \cdot \mathsf{pk} \bmod q)$

Fig. 4. Lattice-based t-out-of-n threshold signatures $\mathsf{TSL}[\mathsf{SecSha}]$, where SecSha is a linear secret sharing scheme with small coefficients (see Definition 2). Here, $\mathrm{H}_1 : \{0,1\}^* \to \mathcal{S}_{\mathsf{b}}^\ell$ and $\mathrm{H}_2 : \{0,1\}^* \to \mathcal{S}_{\mathsf{c}}$. Also, T_i denotes the set of shares of party i and $\lambda_j^{lr.\mathsf{SS}}$ denotes the reconstruction coefficient. Also, we remark that, as stated earlier, the system parameter par is implicitly given to all algorithms except Setup.

scheme SecSha. In particular, the secret key $\mathsf{sk} \in R_q^m$ is shared into L secret shares $\{\mathsf{ss}_j\}_{j\in[L]}$, and for each party $i \in [n]$, its secret key share is $\{\mathsf{ss}_j\}_{j\in T_i}$. For a signer set SS where $|SS| \geqslant t$, by the linearity property of SecSha, the secret key can be reconstructed as $\mathsf{sk} \leftarrow \sum_{i\in SS} \sum_{j\in T_i} \lambda_j^{SS} \mathsf{ss}_j$. Although we do not explicitly provide a DKG protocol, we do not see our use of Benaloh and Leichter's secret sharing in place of Shamir's secret sharing to be a barrier in constructing a DKG. For instance, the DKG protocol given in [GKS24] is a possible candidate as it only utilizes the linearity of the secret sharing schemes, which is satisfied by both Shamir's and Benaloh and Leichter's secret sharing.

For the signing protocol, in the offline round, each signer generates $\ell + 1$ nonces $\{\boldsymbol{R}_j\}_{j\in[0..\ell]}$ as a pre-processing token, where $\boldsymbol{R}_j \leftarrow A\boldsymbol{r}_j$ for a uniformly sampled $A \in R_q^{k\times m}$ generated during the setup phase and $\boldsymbol{r}_j$ sampled from the discrete Gaussian distribution $\mathscr{D}^m_{\sigma_{\mathsf{r}}}$. In the online round, given

Parameter	Description
κ	Security parameter
n	Number of parties
t	Threshold for signing
L	Total size of the secret shares
$N \geqslant 2\kappa$	A power of two integer
q	Prime modulus
$R = \mathbb{Z}[X]/(X^N + 1)$	Cyclotomic Ring
$R_q = \mathbb{Z}_q[X]/(X^N + 1)$	Ring
β_{sk}	The ℓ_∞-norm bound of the secret key sk
k	Number of rows of A
$m = (2\kappa/N + k \log q)/\log(2\beta_{\mathsf{sk}})$	Number of columns of A
$\ell + 1 = 2\kappa/\log(2N) + 1$	Number of nonces for each signer
$\mathcal{S}_{\mathsf{b}} = \{\pm 1, \pm X, \ldots, \pm X^{N-1}\}$	Set for the aggregating coefficients b_j
β_{c} chosen such that $2^{\beta_{\mathsf{c}}}\binom{N}{\beta_{\mathsf{c}}} \geqslant 2^{2\kappa}$	The ℓ_1-norm of the challenge c
$\mathcal{S}_{\mathsf{c}} = \{c \in R : \lVert c \rVert_\infty = 1, \lVert c \rVert_1 = \beta_{\mathsf{c}}\}$	Set of the challenges c
$\mathscr{B}_{\beta_{\mathsf{sk}}} = \{s \in R : \lVert s \rVert_\infty \leqslant \beta_{\mathsf{sk}}\}$	Set of elements with bounded ℓ_∞-norm
β_{ss}	Lemma 16's ℓ_∞-norm bound of SecSha
$\sigma_{\mathsf{r}} = \max\{N\sqrt{32\pi \mathsf{q}_s m N}\beta_{\mathsf{c}}\beta_{\mathsf{ss}}\beta_{\mathsf{sk}},$ $\frac{16N\sqrt{3m}}{\sqrt{\pi}} q^{\frac{k}{m}} \sqrt{N(\log(2mN) + 2\kappa)}\}$	Standard deviation of the preimages $\boldsymbol{r}_{i,j}$ for $j \in [0..\ell]$
$\beta_{\mathsf{z}} = \sqrt{mN}(\sqrt{t(\ell+1)}\sigma_{\mathsf{r}} + 2\beta_{\mathsf{c}}\beta_{\mathsf{sk}})$	The ℓ_2-norm bound for the aggregated $\boldsymbol{z}$

Fig. 5. Table showing the parameters for the scheme TSL.

a leader request lr, each signer computes an aggregated nonce $\boldsymbol{R}$ from a list of tokens generated by signers in $lr.SS$ using coefficients $\{b_j \in R\}_{j \in [\ell]}$ output from a hash function H_1 and computes a challenge $c \in R$ from another hash function H_2 as described in the algorithm $\mathsf{CompPar}$. Each signer then returns its partial signature $(\boldsymbol{R}, \boldsymbol{z})$. It is worth noting that we include $\boldsymbol{R}$ in partial signatures for the simplicity of presenting our protocol. In actual implementations, each signer only needs to send back $\boldsymbol{z}$ since the leader can compute $\boldsymbol{R}$ from lr by itself.

We note that our protocol does not achieve identifiable abort, i.e., one cannot identify misbehaving signers if the final signature obtained from aggregating the partial signatures is not valid. However, we point out that one possible fix is to let KeyGen additionally output the commitments of the secret key shares as public information and, during signing, have each signer send an NIZK along with their partial signature proving that the partial signature is computed honestly with respect to the committed key shares and the first round nonces, which allows the leader to verify the correctness of the partial signature. A similar approach can be found in [GKS24]. We also note that, except for the recent work [ENP24], which uses 4 online rounds to achieve robustness, other prior works satisfying robustness/identifiable abort rely on advanced primitives, such as NIZKs [BGG+18, CCK23, GKS24] or homomorphic signatures [ASY22].

Parameters. In Figure 5, we give the description of the parameters used in the protocol. We set ℓ and β_{c} such that the sizes of $\mathcal{S}_{\mathsf{b}}^{\ell}$ and $\mathcal{S}_{\mathsf{c}}$ are at least $2^{2\kappa}$. We set m according to Lemma 20 such that except for a negligible probability, for a secret key uniformly sampled from $\mathscr{B}_{\beta_{\mathsf{sk}}}^m$, there exists another secret key in $\mathscr{B}_{\beta_{\mathsf{sk}}}^m$ such that their corresponding public keys are the same. We set β_{z} according to the correctness proof and σ_{r} according to the unforgeability proof.

Correctness and Unforgeability. The following theorems establish the correctness and unforgeability of TSL. We show correctness in Section 4.4, while we show TS-UF-0 under the MSIS assumption in the random oracle model below.

$\underline{\mathsf{Fork}^{\mathcal{A}}(x):}$
Pick the random coin ρ of $\mathcal{A}$ at random
$(h_1, \ldots, h_q), (\bar{h}_1, \ldots \bar{h}_q) \leftarrow\!\!\$\ \mathsf{HG}$
$(I, \mathrm{Out}) \leftarrow \mathcal{A}(x, h_1, \ldots, h_q; \rho)$
If $I = \perp$ then return $\perp$
$(\bar{I}, \overline{\mathrm{Out}}) \leftarrow \mathcal{A}(x, h_1, \ldots, h_{I-1}, \bar{h}_I, \ldots, \bar{h}_q; \rho)$
If $I \neq \bar{I}$ then return $\perp$
Return $(I, \mathrm{Out}, \overline{\mathrm{Out}})$

Fig. 6. The forking algorithm build from $\mathcal{A}$.

Theorem 1 (Correctness of TSL). *The threshold signature scheme* TSL *is correct with correctness error* $\delta = (2 + 4t(\ell + 1)) \cdot 2^{-2\kappa}$.

Theorem 2 (TS-UF-0 **of** TSL)**.** *For any integers* $q = q(\kappa), k = k(\kappa), m = m(\kappa)$ *and any* TS-UF-0 *adversary* $\mathcal{A}$ *making at most* $\mathsf{q}_s = \mathsf{q}_s(\kappa)$ *queries to* PPO *and* $\mathsf{q}_h = \mathsf{q}_h(\kappa)$ *queries to* RO, *there exists an* MSIS *adversary* $\mathcal{B}$ *running in time roughly two times that of* $\mathcal{A}$ *such that, for any* $\alpha \geqslant 2$,

$$\mathsf{Adv}^{\text{ts-uf-0}}_{\mathsf{TSL}}(\mathcal{A}, \kappa) \leqslant \sqrt{\mathsf{q}\left(2\alpha\delta_\alpha \mathsf{Adv}^{\text{msis}}_{q,k,m,\beta}(\mathcal{B}, \kappa)\right)^{1-\frac{1}{\alpha}} + \mathsf{q}(2 + 8\mathsf{q}^2)2^{-2\kappa}}\ .$$

where $\mathsf{q} = \mathsf{q}_h + \mathsf{q}_s + 1$, $\beta = 2\beta_{\mathsf{z}} + 4\sqrt{mN}\beta_{\mathsf{c}}\beta_{\mathsf{sk}}$, *and* $\delta_\alpha = (1 + 160\ell\mathsf{q} \cdot 2^{-2\kappa}) \cdot e^\alpha$.

To prove the above theorem, we use the following variant of the forking lemma from [BTZ22], which is proved in Appendix B. The only difference is that here each h_i might be sampled independently from a different distribution. We require it in our proof since the ranges of H_1 and H_2 are different.

Lemma 18. *Let* $q \geqslant 1$ *be an integer,* $S \subseteq [1..q]$ *be a set, and* HG *be an algorithm that outputs* $h_1, \ldots, h_q$ *where each* h_i *is independently sampled. Let* $\mathcal{A}$ *be a randomized algorithm that on input* $x, h_1, \ldots, h_q$ *outputs a pair* (I, Out), *where* $I \in \{\perp\} \cup S$ *and* Out *is a side output. Let* IG *be a randomized algorithm that generates* x. *The accepting probability of* $\mathcal{A}$ *is defined as*

$$\mathrm{acc}(\mathcal{A}) = \mathsf{Pr}_{x \leftarrow\$\,\mathsf{IG}, h_1,\ldots,h_q \leftarrow\$\,\mathsf{HG}}[(I, \mathrm{Out}) \leftarrow\!\!\$\ \mathcal{A}(x, h_1, \ldots, h_q)\ :\ I \neq \perp]\ .$$

Consider algorithm $\mathsf{Fork}^{\mathcal{A}}$ *described in Figure 6. The accepting probability of* $\mathsf{Fork}^{\mathcal{A}}$ *is defined as*

$$\mathrm{acc}(\mathsf{Fork}^{\mathcal{A}}) = \mathsf{Pr}_{x \leftarrow\$\,\mathsf{IG}}[\alpha \leftarrow\!\!\$\ \mathsf{Fork}^{\mathcal{A}}(x)\ :\ \alpha \neq \perp]\ .$$

Then, $\mathrm{acc}(\mathsf{Fork}^{\mathcal{A}}) \geqslant \mathrm{acc}(\mathcal{A})^2/|S|$.

Proof (of Theorem 2). Let $\mathcal{A}$ be a TS-UF-0 adversary described in the theorem. W.l.o.g. we assume that $\mathcal{A}$ is deterministic and corrupts exactly $t - 1$ signers. Also, we assume if $\mathcal{A}$ returns $(\mu^*, (\boldsymbol{R}^*, \boldsymbol{z}^*))$, the RO query $\mathrm{H}_2(\mathsf{pk}, \mu^*, \boldsymbol{R}^*)$ was made by $\mathcal{A}$, which adds at most one RO query. Also, since the game makes at most one RO query to H_1 and H_2 respectively for each signing query, the total number of RO queries to each of H_1 and H_2 is bounded $\mathsf{q} = \mathsf{q}_h + \mathsf{q}_s + 1$. We first construct an algorithm $\mathcal{C}$ compatible with the syntax in Lemma 18 and construct $\mathcal{B}$ from $\mathsf{Fork}^{\mathcal{C}}$. The input of $\mathcal{C}$ consists of $par = A$, public key pk, secret key shares $\{\mathsf{sk}_i\}_{i\in[n]}$, the randomness $\{\boldsymbol{r}_j^{(i)}\}_{i\in[\mathsf{q}_s], j\in[0..\ell]}$ for generating the nonces, and the random RO outputs $h_1, \ldots, h_{2\mathsf{q}}$, where $h_{2i-1} \in \mathcal{S}_{\mathsf{b}}^{\ell}$ and $h_{2i} \in \mathcal{S}_{\mathsf{c}}$

for $i \in [\mathsf{q}]$. To start with, $\mathcal{C}$ does initialization exactly as in the game TS-UF-0 and then runs $\mathcal{A}$ with access to oracles Init, PPO, PSignO simulated in the same manner as in the game TS-UF-0 (the randomness $\{\boldsymbol{r}_j^{(i)}\}_{j\in[0..\ell]}$ are used for the i-th signing query to PPO) and the RO oracle $\widetilde{\mathrm{RO}}$, which is simulated as follows.

$\widetilde{\mathrm{RO}}$ query $\mathrm{H}_1(x)$**:** If $\mathrm{H}_1(x) \neq \bot$, $\mathcal{C}$ returns $\mathrm{H}_1(x)$. Otherwise, parse x as $(\widetilde{\mathsf{pk}}, lr)$. If the parsing fails or $\widetilde{\mathsf{pk}} \neq \mathsf{pk}$, $\mathcal{C}$ sets $\mathrm{H}_1(x) \leftarrow\$\ \mathcal{S}_{\mathsf{b}}^{\ell}$ and returns $\mathrm{H}_1(x)$. Otherwise, $\mathcal{C}$ increases ctr_h by 1, sets $\mathrm{H}_1(x) \leftarrow h_{2\mathrm{ctr}_h-1}$. Also, $\mathcal{C}$ computes $\boldsymbol{R} \leftarrow \sum_{i\in lr.\mathsf{SS}}(\boldsymbol{R}_{i,0} + \sum_{j\in[\ell]} b_j \cdot \boldsymbol{R}_{i,j})$, where $(\boldsymbol{R}_{i,j})_{j\in[0..\ell]} \leftarrow lr.\mathsf{PP}(i)$ and $\{b_j\}_{j\in[\ell]} \leftarrow h_{2\mathrm{ctr}_h-1}$. If $\mathrm{H}_2(\mathsf{pk}, lr.\mathsf{msg}, \boldsymbol{R}) = \bot$, $\mathcal{C}$ sets $\mathrm{H}_2(\mathsf{pk}, lr.\mathsf{msg}, \boldsymbol{R}) \leftarrow h_{2\mathrm{ctr}_h}$. Finally, $\mathcal{C}$ returns $\mathrm{H}_1(x)$.

$\widetilde{\mathrm{RO}}$ query $\mathrm{H}_2(x)$**:** If $\mathrm{H}_2(x) \neq \bot$, $\mathcal{C}$ returns $\mathrm{H}_2(x)$. Otherwise, parse x as $(\widetilde{\mathsf{pk}}, \mu, \boldsymbol{R})$. If the parsing fails or $\widetilde{\mathsf{pk}} \neq \mathsf{pk}$, $\mathcal{C}$ sets $\mathrm{H}_2(x) \leftarrow\$\ \mathcal{S}_{\mathsf{c}}$. Otherwise, $\mathcal{C}$ increases ctr_h by 1 and sets $\mathrm{H}_2(x) \leftarrow h_{2\mathrm{ctr}_h}$. Finally, $\mathcal{C}$ returns $\mathrm{H}_2(x)$.

After receiving the output $(\mu^*, (\boldsymbol{R}^*, \boldsymbol{z}^*))$ from $\mathcal{A}$, $\mathcal{C}$ aborts if $\mathcal{A}$ does not win the TS-UF-0 game. Otherwise $\mathcal{C}$ finds the index I such that $\mathrm{H}_2(\mathsf{pk}, \mu^*, \boldsymbol{R}^*)$ is set to h_I during the simulation. By our assumption of $\mathcal{A}$, we know such I must exist. Then, $\mathcal{C}$ returns $(I, \mathrm{Out} = (\mu^*, \boldsymbol{R}^*, \boldsymbol{z}^*))$.

<u>Analysis of $\mathcal{C}$.</u> To use Lemma 18, we define $S := \{2j\}_{j\in[\mathsf{q}]}$ and IG as the algorithm that runs $A \leftarrow\$\ \mathsf{Setup}(1^\kappa)$, $(\mathsf{pk}, \{\mathsf{sk}_i\}_{i\in[n]}) \leftarrow\$\ \mathsf{KeyGen}()$, samples $\boldsymbol{r}_j^{(i)} \leftarrow\$\ \mathscr{D}_{\sigma_{\mathsf{r}}}^m$ for each $i \in [\mathsf{q}_s]$ and $j \in [0..\ell]$, and returns $(A, \mathsf{pk}, \{\mathsf{sk}_i\}_{i\in[n]}, \{\boldsymbol{r}_j^{(i)}\}_{i\in[\mathsf{q}_s],j\in[0..\ell]})$. We define HG as the algorithm that samples $h_1, h_3, \ldots, h_{2\mathsf{q}-1}$ uniformly from $\mathcal{S}_{\mathsf{b}}^{\ell}$ and $h_2, h_4, \ldots, h_{2\mathsf{q}}$ uniformly from $\mathcal{S}_{\mathsf{c}}$. From the simulation, we know that the output index I of $\mathcal{C}$ is always in S. Also, we can see that $\mathcal{C}$ simulates the game TS-SUF-0 perfectly, which implies $\mathrm{acc}(\mathcal{C}) \geqslant \mathsf{Adv}_{\mathsf{TSL}}^{\text{ts-uf-0}}(\mathcal{A}, \kappa)$. By Lemma 18, we have that

$$\mathrm{acc}(\mathsf{Fork}^{\mathcal{C}}) \geqslant \mathsf{Adv}_{\mathsf{TSL}}^{\text{ts-uf-0}}(\mathcal{A}, \kappa)^2/\mathsf{q}\ .$$

<u>Construct $\mathcal{B}$ from $\mathsf{Fork}^{\mathcal{C}}$.</u> We now construct the MSIS adversary $\mathcal{B}$ using $\mathsf{Fork}^{\mathcal{C}}$. To start with, $\mathcal{B}$ receives $A \in R_q^{k\times m}$ from the MSIS game, follows the algorithm $\mathsf{KeyGen}()$ to generate $(\mathsf{pk}, \mathsf{sk}, \{\mathsf{sk}_i\}_{i\in[n]})$, and samples $\{\boldsymbol{r}_j^{(i)}\}_{i\in[\mathsf{q}_s],j\in[0..\ell]}$ exactly as in IG. Then, $\mathcal{B}$ runs $\mathsf{Fork}^{\mathcal{C}}$. If $\mathsf{Fork}^{\mathcal{C}}$ outputs $(I, \mathrm{Out} = (\mu^*, \boldsymbol{R}^*, \boldsymbol{z}^*), \overline{\mathrm{Out}} = (\bar{\mu}^*, \bar{\boldsymbol{R}}^*, \bar{\boldsymbol{z}}^*))$, $\mathcal{B}$ returns $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk}$. Otherwise, $\mathcal{B}$ aborts.

By the execution of $\mathsf{Fork}^{\mathcal{C}}$, we know $(\mu^*, \boldsymbol{R}^*) = (\bar{\mu}^*, \bar{\boldsymbol{R}}^*)$, $A\boldsymbol{z}^* = \boldsymbol{R}^* + 2h_I \cdot \mathsf{pk}$ and $A\bar{\boldsymbol{z}}^* = \bar{\boldsymbol{R}}^* + 2\bar{h}_I \cdot \mathsf{pk}$. Therefore, $A(\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk}) = 0$. Also, it is clear that $\|\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk}\| \leqslant 2\beta_{\mathsf{z}} + 2\sqrt{mN}\,\|(h_I - \bar{h}_I)\mathsf{sk}\|_\infty \leqslant 2\beta_{\mathsf{z}} + 4\sqrt{mN}\beta_{\mathsf{c}}\beta_{\mathsf{sk}} = \beta$, where the last inequality is due to

$$\|(h_I - \bar{h}_I)\mathsf{sk}\|_\infty \leqslant 2\beta_{\mathsf{c}}\beta_{\mathsf{sk}}\ .$$

It is left to show that $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk} \neq 0$ with high probability. Denote Win as the event that $\mathcal{B}$ returns and $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk} \neq 0$, which means that $\mathcal{B}$ wins the MSIS game, and Zero as the event that $\mathcal{B}$ returns and $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk} = 0$. Since $\mathcal{B}$ returns if $\mathsf{Fork}^{\mathcal{C}}$ returns,

$$\Pr[\mathsf{Win}\ \vee\ \mathsf{Zero}] = \mathrm{acc}(\mathsf{Fork}^{\mathcal{C}}) \geqslant \mathsf{Adv}_{\mathsf{TSL}}^{\text{ts-uf-0}}(\mathcal{A}, \kappa)^2/\mathsf{q}\ . \tag{1}$$

Denote $\mathsf{BadHash}$ as the event that $h_1, \bar{h}_1, \ldots, h_{2\mathsf{q}}, \bar{h}_{2\mathsf{q}}$ are *not* all distinct, $\mathcal{S}_{\mathsf{gA}}$ as the set of MSIS challenge $A \in R_q^{k\times m}$ that $\eta_\varepsilon(\Lambda_q^\perp(A)) \leqslant \sigma_{\mathsf{r}}/(2N\sqrt{3m})$, and $\mathcal{S}_{\mathsf{gk},A}$ as the set of secret key $\mathsf{sk} \in \mathscr{B}_{\beta_{\mathsf{sk}}}^m$ such that there exists another key $\mathsf{sk}' \neq \mathsf{sk}$ and $A\mathsf{sk}' = A\mathsf{sk}$. Then, denote the event Good as $(\neg\mathsf{BadHash}\ \wedge\ A \in \mathcal{S}_{\mathsf{gA}}\ \wedge\ \mathsf{sk} \in \mathcal{S}_{\mathsf{gk},A})$. We bound $\Pr[\mathsf{Win}]$ using the following main lemma proved in Section 4.3.

Lemma 19. *For any $\alpha \geqslant 2$,*

$$\mathsf{Pr}[\mathsf{Win} \ \wedge \ \mathsf{Good}] \geqslant \mathsf{Pr}[\mathsf{Zero} \ \wedge \ \mathsf{Good}]^{\alpha/(\alpha-1)}/\delta_\alpha ,$$

where $\delta_\alpha = (1 + 160\ell \mathsf{q} \cdot 2^{-2\kappa}) \cdot e^\alpha$.

We now show that Good occurs with overwhelming probability. By Lemma 4, $\mathsf{Pr}[A \notin \mathcal{S}_{\mathsf{gA}}] \leqslant 2^{-N} \leqslant 2^{-2\kappa}$. Since each of $h_1, h_3, \ldots, h_{2\mathsf{q}-1}$ and $h_2, h_4, \ldots, h_{2\mathsf{q}}$ are sampled uniformly from $\mathcal{S}_{\mathsf{b}}^\ell$ and $\mathcal{S}_{\mathsf{c}}$ respectively, $\mathsf{Pr}[\mathsf{BadHash}] \leqslant (2\mathsf{q})^2/|\mathcal{S}_{\mathsf{b}}|^\ell + (2\mathsf{q})^2/|\mathcal{S}_{\mathsf{c}}| \leqslant 8\mathsf{q}^2 2^{-2\kappa}$. Also, by the following lemma, $\mathsf{Pr}[\mathsf{sk} \notin \mathcal{S}_{\mathsf{gk},A}] \leqslant 2^{-2\kappa}$.

Lemma 20. *For any $A \in R_q^{k\times m}$ and β_{sk}, if $m \geqslant (2\kappa/N + k\log q)/\log(2\beta_{\mathsf{sk}})$, we have that for $\mathsf{sk} \leftarrow_\$ \mathscr{B}_{\beta_{\mathsf{sk}}}^m$, with probability at least $1 - 2^{-2\kappa}$, there exists $\mathsf{sk}' \in \mathscr{B}_{\beta_{\mathsf{sk}}}^m$ such that $\mathsf{sk} \neq \mathsf{sk}'$ and $A\mathsf{sk} = A\mathsf{sk}'$.*

Proof. Here, one only has to show that the size of $\mathscr{B}_{\beta_{\mathsf{sk}}}^m$ is much larger than R_q^k. Since there is at most q^{kN} possible values of $A\mathsf{sk}$, with probability at most $q^{kN}/(2\beta_{\mathsf{sk}})^{mN}$, the sampled sk would not satisfy the condition in the lemma. Thus, with $m \geqslant 2\kappa/N \log 2\beta_{\mathsf{sk}} + k\log q/\log 2\beta_{\mathsf{sk}}$, the statement is true. □

Therefore, $\mathsf{Pr}[\neg\mathsf{Good}] \leqslant (2 + 8\mathsf{q}^2)2^{-2\kappa}$. Finally, by Lemma 19 and Equation (1), we conclude our theorem, since

$$\begin{aligned}
\mathsf{Pr}[\mathsf{Win}] &\geqslant \frac{1}{2}\left(\mathsf{Pr}[\mathsf{Win} \ \wedge \ \mathsf{Good}] + \mathsf{Pr}[\mathsf{Zero} \ \wedge \ \mathsf{Good}]^{\alpha/(\alpha-1)}/\delta_\alpha\right) \\
&\geqslant \frac{\alpha-1}{2\alpha\delta_\alpha}(\mathsf{Pr}[\mathsf{Win} \ \wedge \ \mathsf{Good}] + \mathsf{Pr}[\mathsf{Zero} \ \wedge \ \mathsf{Good}])^{\alpha/(\alpha-1)} \\
&\geqslant \frac{1}{2\alpha\delta_\alpha}(\mathsf{Pr}[(\mathsf{Win} \ \vee \ \mathsf{Zero}) \ \wedge \ \mathsf{Good}])^{\alpha/(\alpha-1)} \\
&\geqslant \frac{1}{2\alpha\delta_\alpha}\left(\mathsf{Adv}_{\mathsf{TSL}}^{\text{ts-uf-0}}(\mathcal{A},\kappa)^2/\mathsf{q} - (2 + 8\mathsf{q}^2)2^{-2\kappa}\right)^{\alpha/(\alpha-1)} ,
\end{aligned}$$

where the third inequality is due to Lemma 15 and the fact that $\delta_\alpha > 1$. □

4.3 Proof of Lemma 19

By the definition of $\mathcal{S}_{\mathsf{gk},A}$, there exists a bijection $f_A : \mathcal{S}_{\mathsf{gk},A} \to \mathcal{S}_{\mathsf{gk},A}$ such that $f_A(\mathsf{sk}) \neq \mathsf{sk}$ and $A \cdot f(\mathsf{sk}) = A \cdot \mathsf{sk}$. Denote a random variable $T_{A,\mathsf{sk},\boldsymbol{h}}$ as the view of $\mathcal{A}$ during its interaction with $\mathcal{B}$ given the MSIS challenge being A, the secret key being sk and the hash values being $\boldsymbol{h} = (h_1, \ldots, h_{2\mathsf{q}_h}, \bar{h}_1, \ldots, \bar{h}_{2\mathsf{q}_h})$ for answering RO queries. More concretely, $T_{A,\mathsf{sk},\boldsymbol{h}}$ contains the public key pk, the secret key shares of corrupted signers $\{\mathsf{sk}_j\}_{j\in CS}$, the transcripts of all queries to the oracles PPO, PSignO, RO, and the outputs of $\mathcal{A}$ in both executions. Denote $W_{A,\mathsf{sk},\boldsymbol{h}}$ as the distribution of $T_{A,\mathsf{sk},\boldsymbol{h}}$. Denote $\mathcal{S}_{\mathsf{gh}}$ as the set of hash values $\boldsymbol{h}$ such that $\mathsf{BadHash}$ does not occur.

We first show that the lemma holds *if* $R_\alpha\left(W_{A,\mathsf{sk},\boldsymbol{h}} \| W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}\right) \leqslant \delta_\alpha$ *for any* $A \in \mathcal{S}_{\mathsf{gA}}$, $\mathsf{sk} \in \mathcal{S}_{\mathsf{gk},A}$ *and* $\boldsymbol{h} \in \mathcal{S}_{\mathsf{gh}}$. Given a view T, we denote $(\mu^*, \boldsymbol{R}^*, \boldsymbol{z}^*)$ and $(\bar{\mu}^*, \bar{\boldsymbol{R}}^*, \bar{\boldsymbol{z}}^*)$ as the outputs of $\mathcal{A}$ in T, and we follow the execution of $\mathcal{C}$ to find an index I such that $\mathrm{H}_2(\mathsf{pk}, \mu^*, \boldsymbol{R}^*)$ is set to h_I if $\mathcal{A}$ wins during the first execution. Denote $\bar{I}$ as such an index for the second execution of $\mathcal{A}$. We define the event E_{x} as $\mathcal{A}$ wins in both executions and $I = \bar{I} \ \wedge \ \boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{x} = 0$.

For any fixed $A \in \mathcal{S}_{\mathsf{gA}}, \mathsf{sk} \in \mathcal{S}_{\mathsf{gk},A}, \boldsymbol{h} \in \mathcal{S}_{\mathsf{gh}}$ and $T \leftarrow_{\$} W_{A,\mathsf{sk},\boldsymbol{h}}$, if E_{sk} occurs, since $\mathsf{sk} \neq f_A(\mathsf{sk})$ and $\boldsymbol{h} \in \mathcal{S}_{\mathsf{gh}}$ which implies $h_I - \bar{h}_I \neq 0$, we know $\boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I) f_A(\mathsf{sk}) \neq \boldsymbol{z}^* - \bar{\boldsymbol{z}}^* - 2(h_I - \bar{h}_I)\mathsf{sk} = 0$, which means that $\mathcal{B}$ wins the MSIS game given $(A, f_A(\mathsf{sk}), \boldsymbol{h}, T)$. Therefore, $\Pr[\mathsf{Win}|A, f_A(\mathsf{sk}), \boldsymbol{h}] \geqslant \Pr_{T \leftarrow_{\$} W_{A,\mathsf{sk},\boldsymbol{h}}}[\mathsf{E}_{\mathsf{sk}}]$, where $\Pr[\mathsf{Win}|A, f_A(\mathsf{sk}), \boldsymbol{h}]$ denotes the probability that Win occurs given the MSIS challenge being A, the secret key being $f_A(\mathsf{sk})$, and the hash values being $\boldsymbol{h}$. For $T \leftarrow_{\$} W_{A,\mathsf{sk},\boldsymbol{h}}$, if E_{sk} occurs, we know the event Zero occurs given the secret key being sk and the view of $\mathcal{A}$ being T, which means $\Pr[\mathsf{Zero}|A, \mathsf{sk}, \boldsymbol{h}] = \Pr_{T \leftarrow_{\$} W_{A,\mathsf{sk},\boldsymbol{h}}}[\mathsf{E}_{\mathsf{sk}}]$. Therefore, by Lemma 6,

$$\begin{aligned}
\Pr[\mathsf{Win}|A, f_A(\mathsf{sk}), \boldsymbol{h}] &\geqslant \Pr_{T \leftarrow_{\$} W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}}[\mathsf{E}_{\mathsf{sk}}] \\
&\geqslant \Pr_{T \leftarrow_{\$} W_{A,\mathsf{sk},\boldsymbol{h}}}[\mathsf{E}_{\mathsf{sk}}]^{\alpha/(\alpha-1)} / R_\alpha \left(W_{A,\mathsf{sk},\boldsymbol{h}} \| W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}\right) \\
&\geqslant \Pr[\mathsf{Zero}|A, \mathsf{sk}, \boldsymbol{h}]^{\alpha/(\alpha-1)} / \delta_\alpha \ ,
\end{aligned}$$

which implies

$$\begin{aligned}
\Pr[\mathsf{Win}|\mathsf{Good}] &= \mathop{\mathbb{E}}_{A,\mathsf{sk},\boldsymbol{h}}[\Pr[\mathsf{Win}|A, f_A(\mathsf{sk}), \boldsymbol{h}]] \\
&\geqslant \mathop{\mathbb{E}}_{A,\mathsf{sk},\boldsymbol{h}}[\Pr[\mathsf{Zero}|A, \mathsf{sk}, \boldsymbol{h}]^{\alpha/(\alpha-1)} / \delta_\alpha] \\
&\geqslant \mathop{\mathbb{E}}_{A,\mathsf{sk},\boldsymbol{h}}[\Pr[\mathsf{Zero}|A, \mathsf{sk}, \boldsymbol{h}]]^{\alpha/(\alpha-1)} / \delta_\alpha \\
&= \Pr[\mathsf{Zero}|\mathsf{Good}]^{\alpha/(\alpha-1)} / \delta_\alpha \ ,
\end{aligned}$$

where the expectation is taken over $(A, \mathsf{sk}, \boldsymbol{h})$ uniformly sampled from $\mathcal{S}_{\mathsf{gA}} \times \mathcal{S}_{\mathsf{gk},A} \times \mathcal{S}_{\mathsf{gh}}$, the first equation is due to the fact that f_A is a bijection on $\mathcal{S}_{\mathsf{gk},A}$, and the last inequality is due to Jensen's inequality. Therefore,

$$\begin{aligned}
\Pr[\mathsf{Win} \ \wedge \ \mathsf{Good}] = \Pr[\mathsf{Win}|\mathsf{Good}]\Pr[\mathsf{Good}] \geqslant \frac{1}{\delta_\alpha}\Pr[\mathsf{Good}] \cdot \Pr[\mathsf{Zero}|\mathsf{Good}]^{\alpha/(\alpha-1)} \\
\geqslant \frac{1}{\delta_\alpha}(\Pr[\mathsf{Good}] \cdot \Pr[\mathsf{Zero}|\mathsf{Good}])^{\alpha/(\alpha-1)} = \frac{1}{\delta_\alpha}(\Pr[\mathsf{Zero} \ \wedge \ \mathsf{Good}])^{\alpha/(\alpha-1)} \ ,
\end{aligned}$$

where the second inequality is due to $\Pr[\mathsf{Good}] \leqslant 1$ and $\frac{\alpha}{\alpha-1} > 1$.

ANALYSIS OF $R_\alpha \left(W_{A,\mathsf{sk},\boldsymbol{h}} \| W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}\right)$. We first define a more fine-grained view $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ by further fixing the randomness $\boldsymbol{\rho}$ used for generating the secret key shares. We can view $W_{A,\mathsf{sk},\boldsymbol{h}}$ as the distribution of $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ for $\boldsymbol{\rho}$ uniformly sampled from $(R_q^m)^K$.

We also extend the bijection f_A to a bijection f'_A that additionally takes the randomness $\boldsymbol{\rho}$ as input such that f'_A maps $(\mathsf{sk}, \boldsymbol{\rho})$ to $(f_A(\mathsf{sk}), \boldsymbol{\rho}')$ such that the shares of corrupted signers CS given $(\mathsf{sk}, \boldsymbol{\rho})$ are the same as that given $(f_A(\mathsf{sk}), \boldsymbol{\rho}')$.[6] By Lemma 16, we construct the bijection as $f'_A(\mathsf{sk}, \boldsymbol{\rho}) := (f_A(\mathsf{sk}), \Phi_{\mathsf{sk}, f_A(\mathsf{sk}), CS}(\boldsymbol{\rho}))$. As a result, $W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}$ can be viewed as the distribution of $T_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}}$ for uniformly sampled $\boldsymbol{\rho}$.

Denote $W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ as the distribution of $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$. Denote P as the distribution of $(\boldsymbol{\rho}, T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}})$ and Q as the distribution of $(\boldsymbol{\rho}, T_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}})$ for uniformly sampled $\boldsymbol{\rho}$. By the data processing inequality from Lemma 6, we have that $R_\alpha \left(W_{A,\mathsf{sk},\boldsymbol{h}} \| W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}\right) \leqslant R_\alpha (P \| Q)$. By Lemma 7, denoting P_1 as

[6] The corrupted set CS is fixed here since we assume that $\mathcal{A}$ is deterministic.

the uniform distribution of $\boldsymbol{\rho}$ and $P_{2|\boldsymbol{\rho}}$ as the distribution of $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ conditioned on the value of $\boldsymbol{\rho}$ (Q_1 and $Q_{2|\boldsymbol{\rho}}$ are defined analogously), then

$$R_\alpha\left(P\|Q\right) \leqslant R_\alpha\left(P_1\|Q_1\right) \cdot \max_{\boldsymbol{\rho}} R_\alpha\left(P_{2|\boldsymbol{\rho}}\|Q_{2|\boldsymbol{\rho}}\right)$$
$$= \max_{\boldsymbol{\rho}} R_\alpha\left(W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}\|W_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}}\right) .$$

Therefore,

$$R_\alpha\left(W_{A,\mathsf{sk},\boldsymbol{h}}\|W_{A,f_A(\mathsf{sk}),\boldsymbol{h}}\right) \leqslant \max_{\boldsymbol{\rho}} R_\alpha\left(W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}\|W_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}}\right) ,$$

and we can conclude the lemma by the following claim.

Claim. For any $A \in \mathcal{S}_{\mathsf{gA}}$, $\mathsf{sk} \in \mathcal{S}_{\mathsf{gk},A}$, $\boldsymbol{\rho} \in (R_q^m)^K$, and $\boldsymbol{h} \in \mathcal{S}_{\mathsf{gh}}$,

$$R_\alpha(W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}\|W_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}}) \leqslant (1 + 160\ell\mathsf{q} \cdot 2^{-2\kappa}) \cdot e^\alpha .$$

Proof. Denote $(\mathsf{sk}', \boldsymbol{\rho}') = f'_A(\mathsf{sk}, \boldsymbol{\rho})$ and denote $\{\mathsf{ss}_i\}_{i\in[L]}$ and $\{\mathsf{ss}'_i\}_{i\in[L]}$ as the secret shares generated by $\mathsf{SecSha.Share}(\mathsf{sk}; \boldsymbol{\rho})$ and $\mathsf{SecSha.Share}(\mathsf{sk}'; \boldsymbol{\rho}')$, respectively. Since $\mathcal{A}$ is deterministic, $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ is determined by the nonces $\{\boldsymbol{R}_0^{(j)}, \ldots, \boldsymbol{R}_\ell^{(j)}\}_{j\in[\mathsf{q}_s]}$ and the outputs $(\boldsymbol{R}, \boldsymbol{z})$ of queries to oracle PSignO. Therefore, we only need to consider the marginal distribution of those variables when comparing the two distributions. We further ignore $\boldsymbol{R}$ from the outputs of PSignO queries since it is determined given $\{\boldsymbol{R}_0^{(j)}, \ldots, \boldsymbol{R}_\ell^{(j)}\}_{j\in[\mathsf{q}_s]}$ and $\boldsymbol{h}$.

We now use Lemma 7 to bound $R_\alpha(W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}\|W_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}})$ by defining random variables $X_0, \ldots, X_{2\mathsf{q}_s}$ as follows. Let $X_0 := \{\boldsymbol{R}_0^{(j)}, \ldots, \boldsymbol{R}_\ell^{(j)}\}_{j\in[\mathsf{q}_s]}$. For $j \in [\mathsf{q}_s]$, let X_j be the output $\boldsymbol{z}$ of the j-th query to PSignO made by $\mathcal{A}$ during the first execution, and let X_j be $\perp$ if $\mathcal{A}$ makes less than j queries to PSignO during the first execution. Similarly, let X_{q_s+j} be the output $\boldsymbol{z}$ of the j-th query to PSignO made by $\mathcal{A}$ during the second execution, and let X_{q_s+j} be $\perp$ if $\mathcal{A}$ does not win during the first execution or makes less than j queries to PSignO during the second execution. We denote D and D' as the distributions of $X_0, \ldots, X_{2\mathsf{q}_s}$ sampled from $W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ and $W_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}}$, respectively.

By Lemma 7, denoting $D_{j|x_{[0..j-1]}}$ as the distribution of X_j conditioned on $x_{[0..j-1]}$ ($D'_{j|x_{[0..j-1]}}$ is defined analogously), we only need to upper-bound $R_\alpha(D_{j|x_{[0..j-1]}}\|D'_{j|x_{[0..j-1]}})$ for any $j \in [0..2\mathsf{q}_s]$ and $x_{[0..j-1]}$. For simplicity of our explanation, we denote

$$\delta_{\alpha,j} := \max_{x_{[0..j-1]}} R_\alpha(D_{j|x_{[0..j-1]}}\|D'_{j|x_{[0..j-1]}}) .$$

For $j = 0$, since $\{\boldsymbol{R}_0^{(j)}, \ldots, \boldsymbol{R}_\ell^{(j)}\}_{j\in[\mathsf{q}_s]}$ are sampled independently of $\mathsf{sk}, \boldsymbol{\rho}$, D_0 and D'_0 are the same distributions, which implies $\delta_{\alpha,j} = 1$.

For $1 \leqslant j \leqslant \mathsf{q}_s$, given $X_{[0..j-1]} = x_{[0..j-1]}$ for any $x_{[0..j-1]}$, we know $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ and $T_{A,f'_A(sk,\boldsymbol{\rho}),\boldsymbol{h}}$ are identical prior to the j-th PSignO query in the first execution. Denote the j-th query to PSignO as (i, lr). We say that the query corresponds to the j'-th token if $lr.\mathrm{PP}(i) = (\boldsymbol{R}_0^{(j')}, \ldots, \boldsymbol{R}_\ell^{(j')})$. Suppose that the query is valid, i.e., the query corresponds to the j'-th token for some $j' \in [\mathsf{q}_s]$ and there is no prior PSignO query corresponding to the same token. Let $(c, b_1, \ldots, b_\ell)$ be the parameters computed from $\mathsf{CompPar}(\mathsf{pk}, lr)$. Let $\boldsymbol{s}_{\hat{i}} \in R^m$ be an arbitrary vector such that $A\boldsymbol{s}_{\hat{i}} = \boldsymbol{R}_{\hat{i}}^{(j')}$ for $\hat{i} \in [0..\ell]$. Then, the distribution of $\boldsymbol{r}_{\hat{i}}^{(j')}$ given $\boldsymbol{R}_{\hat{i}}^{(j')}$ is $\mathscr{D}^m_{\Lambda_q^\perp(A)+\boldsymbol{s}_{\hat{i}},\sigma_\mathsf{r}}$ for $\hat{i} \in [0..\ell]$. Let $\boldsymbol{v} := \sum_{\hat{j}\in T_i} \lambda_{\hat{j}}^{lr.SS} \mathsf{ss}_{\hat{j}}$.

Since $X_j = \boldsymbol{r}_0^{(j')} + \sum_{\hat{i}\in[\ell]} b_{\hat{i}} \boldsymbol{r}_{\hat{i}}^{(j')} + 2c\boldsymbol{v}$ and $\eta_\varepsilon(\Lambda_q^\perp(A)) \leqslant \sigma_\mathsf{r}/(2\sqrt{3mN})$, by Lemma 11, we have $D_j \overset{\varepsilon'}{\approx} \mathscr{D}^{m, \bmod q}_{\Lambda_q^\perp(A)+\boldsymbol{S}+2c\boldsymbol{v},\sigma',2c\boldsymbol{v}}$,[7] where $\varepsilon' = \frac{2((1+\varepsilon)^\ell-1)}{2-(1+\varepsilon)^\ell}$, $\boldsymbol{S} = \boldsymbol{s}_0 + \sum_{\hat{i}\in[\ell]} b_{\hat{i}}\boldsymbol{s}_{\hat{i}}$ and $\sigma' = \sigma_\mathsf{r}\sqrt{1+\ell}$. Similarly, $D'_j \overset{\varepsilon'}{\approx} \mathscr{D}^{m, \bmod q}_{\Lambda_q^\perp(A)+\boldsymbol{S}+2c\boldsymbol{v}',\sigma',2c\boldsymbol{v}'}$, where $\boldsymbol{v}' = \sum_{\hat{j}\in T_i} \lambda_{\hat{j}}^{lr.SS} \mathsf{ss}'_{\hat{j}}$. By weak triangle inequality from Lemma 6,

$$\delta_{\alpha,j} \leqslant (1+\varepsilon')^{1+\frac{\alpha}{\alpha-1}} R_\alpha \left(\mathscr{D}^{m, \bmod q}_{\Lambda_q^\perp(A)+\boldsymbol{S}+2c\boldsymbol{v},\sigma',2c\boldsymbol{v}} \middle\| \mathscr{D}^{m, \bmod q}_{\Lambda_q^\perp(A)+\boldsymbol{S}+2c\boldsymbol{v}',\sigma',2c\boldsymbol{v}'} \right) .$$

By Lemma 16, we have $A\boldsymbol{v} = A\boldsymbol{v}'$, which implies $2c(\boldsymbol{v}-\boldsymbol{v}') \in \Lambda_q^\perp(A)$, and thus the two lattice cosets $\Lambda_q^\perp(A) + \boldsymbol{S} + 2c\boldsymbol{v}$ and $\Lambda_q^\perp(A) + \boldsymbol{S} + 2c\boldsymbol{v}'$ are the same. Then, by Lemma 16, we have $\|\boldsymbol{v} - \boldsymbol{v}'\| \leqslant \beta_\mathsf{ss}\beta_\mathsf{sk}\sqrt{Nm}$, so $\|2c(\boldsymbol{v}-\boldsymbol{v}')\|^2 \leqslant 4\beta_\mathsf{c}^2\beta_\mathsf{ss}^2\beta_\mathsf{sk}^2 Nm$. Thus, by Lemma 9,

$$\delta_{\alpha,j} \leqslant (1+\varepsilon')^{1+\frac{\alpha}{\alpha-1}} \exp\left(\alpha\pi \frac{\|2c(\boldsymbol{v}-\boldsymbol{v}')\|^2}{\sigma'^2}\right) \leqslant (1+\varepsilon')^3 e^{\alpha/(2\mathsf{q})} , \tag{2}$$

where the last inequality is due to the fact that σ_r is set as shown in Figure 5. If the j-query is not valid or $\mathcal{A}$ makes less than j queries to PSignO in the first execution, we have $X_j = \perp$ in both distributions, which means $\delta_{\alpha,j} = 1$.

For $\mathsf{q}_s + 1 \leqslant j \leqslant 2\mathsf{q}_s$, given $X_{[0..j-1]} = x_{[0..j-1]}$ for any $x_{[0..j-1]}$, we know $T_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}}$ and $T_{A,f'_A(sk,\boldsymbol{\rho}),\boldsymbol{h}}$ are identical prior to the $(j-\mathsf{q}_s)$-th PSignO query in the second execution. W.l.o.g. assume $\mathcal{A}$ wins the TS-UF-0 game during the first execution since otherwise $X_j = \perp$ in both D_j and D'_j and $\delta_{\alpha,j} = 1$. Also, w.l.o.g. assume $\mathcal{A}$ makes at least $(j-\mathsf{q}_s)$ queries to PSignO and the $(j-\mathsf{q}_s)$-th query is valid during the second execution since otherwise $X_j = \perp$. We denote the $(j-\mathsf{q}_s)$-th query as $(i, \bar{lr})$ and let $(\bar{c}, \bar{b}_1, \dots, \bar{b}_\ell)$ be the parameters computed from $\mathsf{CompPar}(\mathsf{pk}, \bar{lr})$. Suppose the query corresponds to the j'-th token. There are three cases:

- The adversary does not make a PSignO query that corresponds to the j'-th token during the first execution. Since X_j is the distribution of $\bar{z}$ conditioning on $\{\boldsymbol{R}_0^{(j')}, \dots, \boldsymbol{R}_\ell^{(j')}\}$, we can use the same analysis as in the case of the first execution and get the same bound on $\delta_{\alpha,j}$ as Equation (2).
- Otherwise, the adversary makes a valid PSignO query that also corresponds to the j'-th token during the first execution. Denote the query as (i, lr), and suppose it is the $\tilde{j}$-th PSignO query. (Since the query corresponds to the j'-th token, it must be for signer i too.) Let $(c, b_1, \dots, b_\ell)$ be the parameters computed from $\mathsf{CompPar}(\mathsf{pk}, lr)$ during the first execution. Denote J as the index such that $(b_1, \dots, b_\ell) = h_J$. If $lr = \bar{lr}$ and $J < I$, where we recall that I denotes the index such that $\mathrm{H}_2(\mathsf{pk}, \mu^*, \boldsymbol{R}^*) = h_I$ and $(\mu^*, (\boldsymbol{R}^*, \boldsymbol{z}^*))$ denotes the output of A during the first execution, we have $(\bar{b}_1, \dots, \bar{b}_\ell) = h_J$.
  Denote J' as the index such that $c = h_{J'}$. By the simulation of the random oracles, J' is either $J+1$ or less than J. Since $\mathcal{A}$ wins the TS-UF-0 game during the first execution, $\mu^* \neq lr.\mathsf{msg}$, which implies $J' \neq I$. Since $J' \leqslant J+1 \leqslant I$, we know $J' < I$. Therefore, from the algorithm $\mathsf{CompPar}$, we know $c = h_{J'} = \bar{c}$, which implies that the answer to the $(j-\mathsf{q})$-th PSignO query during the second execution is the same as the $\tilde{j}$-th PSignO query during the first execution. Thus, $X_j = X_{\tilde{j}} = x_{\tilde{j}}$ for both $D_{j|x_{[0..j-1]}}$ and $D'_{j|x_{[0..j-1]}}$ and $\delta_{\alpha,j} = 1$.
- Otherwise, either $lr \neq \bar{lr}$ or $J > I$. Since $\boldsymbol{h} \in \mathcal{S}_\mathsf{gh}$, in either of the cases, $(b_1, \dots, b_\ell) \neq (\bar{b}_1, \dots, \bar{b}_\ell)$. We denote the output of the $\tilde{j}$-th PSignO query during the first execution as $\boldsymbol{z}$ and define

[7] This follows from Lemma 11 showing that $X_j - 2c\boldsymbol{v}$ is distributed closely to $\mathscr{D}^{m, \bmod q}_{\Lambda_q^\perp(A)+\boldsymbol{S},\sigma'}$.

$\{\boldsymbol{s}_{\hat{i}}\}_{\hat{i}\in[0..\ell]}$, and $(\boldsymbol{v},\boldsymbol{v}')$ for the query following the analysis of the first execution. Then, we have that $X_j = \boldsymbol{r}_0 + \sum_{\hat{i}\in[\ell]} \bar{b}_{\hat{i}}\boldsymbol{r}_{\hat{i}} + 2\bar{c}\bar{\boldsymbol{v}}$, where $\bar{\boldsymbol{v}} = \sum_{\hat{j}\in T_i} \lambda_{\hat{j}}^{\bar{lr}.SS}\mathsf{ss}_{\hat{j}}$ and each $\boldsymbol{r}_{\hat{i}}$, for $\hat{i} \in [0..\ell]$, is independently sampled from $\mathscr{D}^{m,\bmod q}_{\Lambda_q^\perp(A)+\boldsymbol{s}_{\hat{i}},\sigma_{\mathsf{r}}}$ conditioning on $\boldsymbol{r}_0 + \sum_{\hat{i}\in[\ell]} b_{\hat{i}}\boldsymbol{r}_{\hat{i}} = \boldsymbol{z} - 2c\boldsymbol{v}$. Denote $\boldsymbol{y}_0 = \boldsymbol{z} - 2c\boldsymbol{v}$. By Lemma 11,

$$D_j \overset{\varepsilon'}{\approx} \mathscr{D}^{m,\bmod q}_{\mathcal{I}\otimes\Lambda_q^\perp(A)+\boldsymbol{y}_0+\boldsymbol{S}'+2\bar{c}\bar{\boldsymbol{v}},\frac{\Delta(\Sigma)}{\Sigma_{11}}\cdot\mathbb{I}_m,\frac{\Sigma_{12}}{\Sigma_{11}}\boldsymbol{y}_0+2\bar{c}\bar{\boldsymbol{v}}}$$

where $\boldsymbol{S}' = \sum_{\hat{i}\in[\ell]}(\bar{b}_{\hat{i}} - b_{\hat{i}})\boldsymbol{s}_{\hat{i}}$, $\mathcal{I}$ denotes the ideal generated by $b_1 - \bar{b}_1, \ldots, b_n - \bar{b}_n$, and $\Sigma = \sigma_{\mathsf{r}}^2 \begin{pmatrix} 1 + \sum_{\hat{i}\in[\ell]} b_{\hat{i}}^* b_{\hat{i}} & 1 + \sum_{\hat{i}\in[\ell]} \bar{b}_{\hat{i}} b_{\hat{i}}^* \\ 1 + \sum_{\hat{i}\in[\ell]} \bar{b}_{\hat{i}}^* b_{\hat{i}} & 1 + \sum_{\hat{i}\in[\ell]} \bar{b}_{\hat{i}}^* \bar{b}_{\hat{i}} \end{pmatrix}$.

Similarly, $D_j' \overset{\varepsilon'}{\approx} \mathscr{D}^{m,\bmod q}_{\mathcal{I}\otimes\Lambda_q^\perp(A)+\boldsymbol{y}_0'+\boldsymbol{S}'+2\bar{c}\bar{\boldsymbol{v}}',\frac{\Delta(\Sigma)}{\Sigma_{11}}\cdot\mathbb{I}_m,\frac{\Sigma_{12}}{\Sigma_{11}}\boldsymbol{y}_0'+2\bar{c}\bar{\boldsymbol{v}}'}$, where

$$\boldsymbol{y}_0' = \boldsymbol{z} - 2c\boldsymbol{v}' \text{ and } \bar{\boldsymbol{v}}' = \sum_{\hat{j}\in T_i} \lambda_{\hat{j}}^{\bar{lr}.SS}\mathsf{ss}_{j'}' \ .$$

Since $(b_1, \ldots, b_\ell) \neq (\bar{b}_1, \ldots, \bar{b}_\ell)$, we know $2 \in \mathcal{I}$ by Lemma 1. Since $c(\boldsymbol{v} - \boldsymbol{v}') + \bar{c}(\bar{\boldsymbol{v}} - \bar{\boldsymbol{v}}') \in \Lambda_q^\perp(A)$ by Lemma 16, we know $2c(\boldsymbol{v}-\boldsymbol{v}')+2\bar{c}(\bar{\boldsymbol{v}}-\bar{\boldsymbol{v}}') \in 2\Lambda_q^\perp(A) \subset \mathcal{I}\otimes\Lambda_q^\perp(A)$, which implies $\mathcal{I}\otimes\Lambda_q^\perp(A) + \boldsymbol{z}-2c\boldsymbol{v}+\boldsymbol{S}'+2\bar{c}\bar{\boldsymbol{v}}$ and $\mathcal{I}\otimes\Lambda_q^\perp(A)+\boldsymbol{z}-2c\boldsymbol{v}'+\boldsymbol{S}'+2\bar{c}\bar{\boldsymbol{v}}'$ are the same lattice cosets. Also, since $b^\dagger b = 1$ for any $b \in \mathcal{S}_{\mathsf{b}}$, we have $\Sigma_{11} = \Sigma_{22} = (1+\ell)\sigma_{\mathsf{r}}^2$. Let $w = \sum_{\hat{i}\in[\ell]} \bar{b}_{\hat{i}} b_{\hat{i}}^*$. Since $\|w\|_1 \leqslant \ell$ and $w \neq \ell$, by Lemma 14, we have $\sigma_{\max}(\Sigma_{12}\Sigma_{21}) = \sigma_{\max}(\sigma_{\mathsf{r}}^4(1+w)^*(1+w)) = \sigma_{\mathsf{r}}^4\sigma_{\max}((1+w)^*(1+w)) \leqslant \sigma_{\mathsf{r}}^4((\ell+1)^2 - 2\ell/N^2)$. Therefore, $\sigma_{\min}(\Delta(\Sigma)) = \sigma_{\min}(\Sigma_{11}\Sigma_{22} - \Sigma_{21}\Sigma_{12}) = \sigma_{\min}((\ell+1)^2\sigma_{\mathsf{r}}^4 - \Sigma_{21}\Sigma_{12}) \geqslant (\ell+1)^2\sigma_{\mathsf{r}}^4 - \sigma_{\max}(\Sigma_{21}\Sigma_{12}) \geqslant 2\ell\sigma_{\mathsf{r}}^4/N^2$, which implies $\sigma_{\min}(\Delta(\Sigma)/\Sigma_{11}) \geqslant \frac{2\ell}{N^2(\ell+1)}\sigma_{\mathsf{r}}^2 \geqslant \sigma_{\mathsf{r}}^2/N^2$. Since $2\Lambda_q^\perp(A) \subset \mathcal{I}\otimes\Lambda_q^\perp(A)$, $\eta_\varepsilon(\mathcal{I}\otimes\Lambda_q^\perp(A)) \leqslant 2\eta_\varepsilon(\Lambda_q^\perp(A)) \leqslant \sigma_{\mathsf{r}}/N \leqslant \sqrt{\sigma_{\min}(\Delta(\Sigma)/\Sigma_{11})}$, by Lemma 9 and using weak-triangle inequality as in the case of the first execution, we have

$$\delta_{\alpha,j} \leqslant (1+\varepsilon')^{1+\frac{\alpha}{\alpha-1}} \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^{\frac{\alpha}{\alpha-1}} \cdot \exp\left(\alpha\pi \frac{N^2\left\|\frac{\Sigma_{12}}{\Sigma_{11}}2c(\boldsymbol{v}-\boldsymbol{v}') + 2\bar{c}(\bar{\boldsymbol{v}}-\bar{\boldsymbol{v}}')\right\|^2}{\sigma_{\mathsf{r}}^2}\right).$$

Also, by Lemma 16, $\|\boldsymbol{v} - \boldsymbol{v}'\| \leqslant \beta_{\mathsf{ss}}\beta_{\mathsf{sk}}\sqrt{mN}$ and $\|\bar{\boldsymbol{v}} - \bar{\boldsymbol{v}}'\| \leqslant \beta_{\mathsf{ss}}\beta_{\mathsf{sk}}\sqrt{mN}$, and since

$$\left\|1 + \sum_{\hat{i}\in[\ell]} \bar{b}_{\hat{i}}^\dagger b_{\hat{i}}\right\|_1 \leqslant 1 + \ell \ ,$$

we know that $\left\|\frac{\Sigma_{21}}{\Sigma_{11}}\right\|_1 \leqslant \frac{\sigma_{\mathsf{r}}^2(\ell+1)}{\sigma_{\mathsf{r}}^2(\ell+1)} \leqslant 1$. Therefore, by how σ_{r} is set in Figure 5,

$$\begin{aligned}\delta_{\alpha,j} &\leqslant (1+\varepsilon')^3 \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2 \cdot \exp\left(\frac{16\alpha\pi\beta_{\mathsf{c}}^2\beta_{\mathsf{ss}}^2\beta_{\mathsf{sk}}^2 mN^3}{\sigma_{\mathsf{r}}^2}\right) \\ &\leqslant (1+\varepsilon')^3 \left(\frac{1+\varepsilon}{1-\varepsilon}\right)^2 \cdot e^{\alpha/(2\mathsf{q})} \ . \end{aligned} \tag{3}$$

Since $\varepsilon = 2^{-2\kappa}$ and $\ell \leqslant 2\kappa$, $\varepsilon' \leqslant 8\ell \cdot 2^{-2\kappa}$ and $\frac{1+\varepsilon}{1-\varepsilon} \leqslant 1 + 4 \cdot 2^{-2\kappa}$. From the above analysis, $R_\alpha\left(D_0 \| D_0'\right) = 1$ and by Equations (2) and (3), for any $j \in [2\mathsf{q}_s]$ and $x_{[0..j-1]}$,

$$R_\alpha\left(D_{j|x_{[0..j-1]}} \| D'_{j|x_{[0..j-1]}}\right) \leqslant (1 + 8\ell \cdot 2^{-2\kappa})^5 e^{\alpha/(2\mathsf{q})} .$$

Thus, by Lemma 7, $R_\alpha(W_{A,\mathsf{sk},\boldsymbol{\rho},\boldsymbol{h}} \| W_{A,f'_A(\mathsf{sk},\boldsymbol{\rho}),\boldsymbol{h}}) \leqslant (1 + 8\ell \cdot 2^{-2\kappa})^{10\mathsf{q}} e^{\alpha} \leqslant (1 + 160\ell\mathsf{q} \cdot 2^{-2\kappa}) \cdot e^{\alpha}$. □

4.4 Proof of Theorem 1 (Correctness of TSL)

Let sk be the secret key and $\mathsf{pk} = A\mathsf{sk}$ be the corresponding public key, denote $\{\mathsf{ss}_j\}_{j \in [L]}$ be the output of the secret sharing algorithm and $\mathsf{sk}_i = \{\mathsf{ss}_j\}_{j \in T_i}$ for each signer $i \in [n]$ denotes its secret key share. To show the correctness of the scheme, we consider any signing interaction with any message μ and any signer set $SS \subseteq [n]$ such that $|SS| \geqslant t$ and a message μ. Then, we have to show the following two points: (1) the aggregated signature $(\boldsymbol{R}, \boldsymbol{z})$ satisfies $A\boldsymbol{z} = \boldsymbol{R} + 2\mathrm{H}_2(\mathsf{pk}, \mu, \boldsymbol{R}) \cdot \mathsf{pk}$, and (2) with overwhelming probability, $\|\boldsymbol{z}\| \leqslant \beta_{\mathsf{z}}$.

For the first point, we start by considering how each $\boldsymbol{z}_i$ for $i \in SS$ is generated. Each signer $i \in SS$ first generates $\boldsymbol{r}_{i,0} \leftarrow\$ \mathscr{D}^m_{\sigma_0}$ and $\boldsymbol{r} \leftarrow\$ \mathscr{D}^m_{\sigma_{\mathsf{r}}}$ for $j \in [\ell]$, and sets $\boldsymbol{R}_{i,j} \leftarrow A\boldsymbol{r}_{i,j}$ for $j \in [0..\ell]$. Then, in the second round on the same leader request lr where $lr.SS = SS$, each signer computes the aggregating coefficients $\{b_j\}_{j \in [\ell]} \leftarrow \mathrm{H}_1(\mathsf{pk}, lr)$ and the challenge $c \leftarrow \mathrm{H}_2(\mathsf{pk}, \mu, \boldsymbol{R})$ where $\boldsymbol{R} \leftarrow \sum_{i \in SS} \boldsymbol{R}_{i,0} + \sum_{j \in [\ell]} b_j \boldsymbol{R}_{i,j}$. Then, the returned response $\boldsymbol{z}_i$ is $\boldsymbol{r}_{i,0} + \sum_{j \in [\ell]} b_j \boldsymbol{r}_{i,j} + 2c \cdot (\sum_{j \in T_i} \lambda_j^{SS} \mathsf{ss}_j)$. Assuming that all the parties are honest, so the aggregated nonce $\boldsymbol{R}$ is the same for all parties. By the linearity property of SecSha, it is easy to see that the aggregated $\boldsymbol{z}$ is

$$\boldsymbol{z} = \sum_{i \in SS} \boldsymbol{z}_i = \sum_{i \in SS} \left(\boldsymbol{r}_{i,0} + \sum_{j \in [\ell]} b_j \boldsymbol{r}_{i,j} \right) + 2c \cdot \mathsf{sk} . \tag{4}$$

Moreover, from Equation (4), we have

$$\begin{aligned} A\boldsymbol{z} &= \sum_{i \in SS} \left(A\boldsymbol{r}_{i,0} + \sum_{j \in [\ell]} b_j A\boldsymbol{r}_{i,j} \right) + 2c \cdot (A\mathsf{sk}) \\ &= \sum_{i \in SS} \boldsymbol{R}_{i,0} + \sum_{j \in [\ell]} b_j \boldsymbol{R}_{i,j} + 2c \cdot \mathsf{pk} \\ &= \boldsymbol{R} + 2\mathrm{H}_2(\mathsf{pk}, \mu, \boldsymbol{R}) \cdot \mathsf{pk} , \end{aligned}$$

showing that the check on $A\boldsymbol{z}$ is satisfied.

It is left to show that with overwhelming probability $\|\boldsymbol{z}\| \leqslant \beta_{\mathsf{z}}$. Let the distribution of $\boldsymbol{z}$ as defined by Equation (4) be $D_{\boldsymbol{z}}$. By Lemma 5, $\eta_\varepsilon(R^m) = \eta_\varepsilon(\mathbb{R}^{mN}) \leqslant \sqrt{\log\left(2mN(1+1/\varepsilon)\right)/\pi} \leqslant \sigma_{\mathsf{r}}/(2\sqrt{3mN})$. Therefore, by Lemma 11, we have that $D_{\boldsymbol{z}} \overset{\varepsilon'}{\approx} \mathscr{D}^m_{\sigma, 2c\mathsf{sk}}$ for $\varepsilon' = \frac{2((1+\varepsilon)^{t(\ell+1)-1}-1)}{2-(1+\varepsilon)^{t(\ell+1)-1}}$ and $\sigma^2 = t(1+\ell)\sigma_{\mathsf{r}}^2$, assuming that $|SS| = t$. Thus, $D_{\boldsymbol{z}}$ has statistical distance $\varepsilon'/2$ from $\mathscr{D}^m_{\sigma, 2c\mathsf{sk}}$. By Lemma 3 and by how σ_{r} is set, the error probability that $\|\boldsymbol{z} - 2c \cdot \mathsf{sk}\| \geqslant \sigma\sqrt{mN}$ is at most $\left(\frac{1+\varepsilon}{1-\varepsilon}\right) 2^{-mN}$. Therefore, we have that

$$\|\boldsymbol{z}\| \leqslant \sigma\sqrt{mN} + \|2 \cdot c \cdot \mathsf{sk}\| \leqslant \sigma_{\mathsf{r}}\sqrt{tmN(1+\ell)} + 2\beta_{\mathsf{c}}\beta_{\mathsf{sk}}\sqrt{mN} \leqslant \beta_{\mathsf{z}} ,$$

n	q	k	m	β_{sk}	σ_{r}	β_{z}	$\lvert\mathsf{pk}\rvert$	$\lvert sig\rvert$	Comm.	$\lvert\{\mathsf{sk}_i\}\rvert/n$
5	2^{106}	7	31	2^{23}	$2^{93.12}$	$2^{103.62}$	47.55KB	258.10KB	1.49MB	610.68KB
32	2^{119}	8	50	2^{18}	$2^{105.06}$	$2^{117.25}$	60.73KB	440.32KB	2.02MB	16.60GB

Fig. 7. The concrete parameters and estimated efficiency for $\kappa = 128$ and $n = 5, 32$. In both cases, we use $(N, \ell, \beta_{\mathsf{c}}) = (512, 26, 64)$. The last second column denotes the communication complexity per signer and the last column denotes the average secret key size.

except with error probability

$$\left(\frac{1+\varepsilon}{1-\varepsilon}\right) 2^{-mN} + \frac{\varepsilon'}{2} \leqslant 2 \cdot 2^{-mN} + 4t(\ell+1) \cdot 2^{-2\kappa} \leqslant (2 + 4t(\ell+1)) \cdot 2^{-2\kappa} = \delta \,.$$

The first inequality follows from $\varepsilon = 2^{-2\kappa}$ and $(1+\varepsilon)^{t(\ell+1)-1} \leqslant 1 + 2t(\ell+1)\varepsilon$, so that $\varepsilon' \leqslant 8t(\ell+1) \cdot 2^{-2\kappa}$ and $(1+\varepsilon)/(1-\varepsilon) \leqslant 1 + 4 \cdot 2^{-2\kappa} \leqslant 2$. Then, the next inequality follows from $N \geqslant 2\kappa$. □

4.5 Concrete instantiation and efficiency analysis

We analyze the concrete efficiency of our protocol in the setting considered by [GKS24], where the security parameter is $\kappa = 128$, the maximum number of signing sessions is $\mathsf{q}_s = 2^{64}$ (following NIST recommendations and used in other related works [PKM+24]), and $n = 5$. We consider arbitrary threshold $1 \leqslant t \leqslant n$ here. We set $N = 512$ and $k = 7$. We set q such that the logarithm of β, the ℓ_2-norm of the short solution, satisfies $\log\beta \leqslant 2\sqrt{kN\log q\log\delta}$, according to [MR09]. We use $\delta = 1.005$ as in [GKS24] so that we get roughly 128-bit security of the MSIS problem. Note that we are not choosing the MSIS parameters according to the concrete bounds of Theorem 2, but rather we are choosing parameters so that MSIS gives 128 bits of security. This follows common practice, and it is justified by the fact that our bound is likely not tight due to the use of the Forking Lemma. We will see that the estimated $\beta \leqslant 2^{104.63}$, so we set $q \geqslant 2^{106}$. We set $\beta_{\mathsf{sk}} = 2^{23}$ and then, according to Figure 5, we set $\beta_{\mathsf{c}} = 64$, $m = 33$, $\ell = 26$. We set $\sigma_{\mathsf{r}} = 2^{93.12}$ due to the first term of the maximum function[8] with $\beta_{\mathsf{ss}} \approx 7200$ by Lemma 17. Then, we set $\beta_{\mathsf{z}} = 2^{103.63}$. By Theorem 2, we have that β is bounded by $2^{104.63}$. Then, our public key size is $\lvert\mathsf{pk}\rvert = kN\log q = 47.55$KB, the signature size is $\lvert sig\rvert = (m+k)N\log q = 258.10$KB, the communication complexity per signer is $((\ell+1)k + m)N\log q = 1.49$MB, and the average secret key size is $\lvert\{\mathsf{sk}_i\}\rvert/n = \lfloor n/2\rceil^{4.3}\log(n)\log(q)mN = 610.68$KB. We summarize the parameters in Figure 7, where we also show the concrete parameters and efficiency for $n = 32$ estimated in the same manner as above.

Acknowledgments

This research was partially supported by NSF grants CNS-2026774, CNS-2154174, a JP Morgan Faculty Award, a CISCO Faculty Award, and a gift from Microsoft.

References

AGHS13. Shweta Agrawal, Craig Gentry, Shai Halevi, and Amit Sahai. Discrete Gaussian leftover hash lemma over infinite domains. In Kazue Sako and Palash Sarkar, editors, *ASIACRYPT 2013, Part I*, volume 8269 of *LNCS*, pages 97–116. Springer, Berlin, Heidelberg, December 2013.

[8] The second term is much smaller given the parameters we set.

ANP23. Benny Applebaum, Oded Nir, and Benny Pinkas. How to recover a secret with $o(n)$ additions. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part I*, volume 14081 of *LNCS*, pages 236–262. Springer, Cham, August 2023.

ASY22. Shweta Agrawal, Damien Stehlé, and Anshu Yadav. Round-optimal lattice-based threshold signatures, revisited. In Mikolaj Bojanczyk, Emanuela Merelli, and David P. Woodruff, editors, *ICALP 2022*, volume 229 of *LIPIcs*, pages 8:1–8:20. Schloss Dagstuhl, July 2022.

BCK+14. Fabrice Benhamouda, Jan Camenisch, Stephan Krenn, Vadim Lyubashevsky, and Gregory Neven. Better zero-knowledge proofs for lattice encryption and their application to group signatures. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014, Part I*, volume 8873 of *LNCS*, pages 551–572. Springer, Berlin, Heidelberg, December 2014.

BCK+22. Mihir Bellare, Elizabeth C. Crites, Chelsea Komlo, Mary Maller, Stefano Tessaro, and Chenzhi Zhu. Better than advertised security for non-interactive threshold signatures. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part IV*, volume 13510 of *LNCS*, pages 517–550. Springer, Cham, August 2022.

Bei96. Amos Beimel. Secure schemes for secret sharing and key distribution. *PhD thesis, Israel Institute of Technology, Technion*, 1996.

BGG+18. Dan Boneh, Rosario Gennaro, Steven Goldfeder, Aayush Jain, Sam Kim, Peter M. R. Rasmussen, and Amit Sahai. Threshold cryptosystems from threshold fully homomorphic encryption. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part I*, volume 10991 of *LNCS*, pages 565–596. Springer, Cham, August 2018.

BGG19. Dan Boneh, Rosario Gennaro, and Steven Goldfeder. Using level-1 homomorphic encryption to improve threshold DSA signatures for bitcoin wallet security. In Tanja Lange and Orr Dunkelman, editors, *LATINCRYPT 2017*, volume 11368 of *LNCS*, pages 352–377. Springer, Cham, September 2019.

BGGK17. Dan Boneh, Rosario Gennaro, Steven Goldfeder, and Sam Kim. A lattice-based universal thresholdizer for cryptographic systems. Cryptology ePrint Archive, Report 2017/251, 2017.

BKP13. Rikke Bendlin, Sara Krehbiel, and Chris Peikert. How to share a lattice trapdoor: Threshold protocols for signatures and (H)IBE. In Michael J. Jacobson Jr., Michael E. Locasto, Payman Mohassel, and Reihaneh Safavi-Naini, editors, *ACNS 13International Conference on Applied Cryptography and Network Security*, volume 7954 of *LNCS*, pages 218–236. Springer, Berlin, Heidelberg, June 2013.

BL90. Josh Cohen Benaloh and Jerry Leichter. Generalized secret sharing and monotone functions. In Shafi Goldwasser, editor, *CRYPTO'88*, volume 403 of *LNCS*, pages 27–35. Springer, New York, August 1990.

BL22. Renas Bacho and Julian Loss. On the adaptive security of the threshold BLS signature scheme. In Heng Yin, Angelos Stavrou, Cas Cremers, and Elaine Shi, editors, *ACM CCS 2022*, pages 193–207. ACM Press, November 2022.

BLT+24. Renas Bacho, Julian Loss, Stefano Tessaro, Benedikt Wagner, and Chenzhi Zhu. Twinkle: Threshold signatures from DDH with full adaptive security. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part I*, volume 14651 of *LNCS*, pages 429–459. Springer, Cham, May 2024.

Bol03. Alexandra Boldyreva. Threshold signatures, multisignatures and blind signatures based on the gap-Diffie-Hellman-group signature scheme. In Yvo Desmedt, editor, *PKC 2003*, volume 2567 of *LNCS*, pages 31–46. Springer, Berlin, Heidelberg, January 2003.

Bop85. Ravi B. Boppana. Amplification of probabilistic boolean formulas. In *26th Annual Symposium on Foundations of Computer Science (sfcs 1985)*, pages 20–29, 1985.

BS23. Katharina Boudgoust and Peter Scholl. Simple threshold (fully homomorphic) encryption from LWE with polynomial modulus. In Jian Guo and Ron Steinfeld, editors, *ASIACRYPT 2023, Part I*, volume 14438 of *LNCS*, pages 371–404. Springer, Singapore, December 2023.

BTT22. Cecilia Boschini, Akira Takahashi, and Mehdi Tibouchi. MuSig-L: Lattice-based multi-signature with single-round online phase. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part II*, volume 13508 of *LNCS*, pages 276–305. Springer, Cham, August 2022.

BTZ22. Mihir Bellare, Stefano Tessaro, and Chenzhi Zhu. Stronger security for non-interactive threshold signatures: BLS and FROST. Cryptology ePrint Archive, Report 2022/833, 2022.

CCK23. Jung Hee Cheon, Wonhee Cho, and Jiseung Kim. Improved universal thresholdizer from iterative shamir secret sharing. Cryptology ePrint Archive, Paper 2023/545, 2023.

CF02. Ronald Cramer and Serge Fehr. Optimal black-box secret sharing over arbitrary Abelian groups. In Moti Yung, editor, *CRYPTO 2002*, volume 2442 of *LNCS*, pages 272–287. Springer, Berlin, Heidelberg, August 2002.

CGG+20. Ran Canetti, Rosario Gennaro, Steven Goldfeder, Nikolaos Makriyannis, and Udi Peled. UC non-interactive, proactive, threshold ECDSA with identifiable aborts. In Jay Ligatti, Xinming Ou, Jonathan Katz, and Giovanni Vigna, editors, *ACM CCS 2020*, pages 1769–1787. ACM Press, November 2020.

CGRS23. Hien Chu, Paul Gerhart, Tim Ruffing, and Dominique Schröder. Practical Schnorr threshold signatures without the algebraic group model. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part I*, volume 14081 of *LNCS*, pages 743–773. Springer, Cham, August 2023.

Che23. Yanbo Chen. DualMS: Efficient lattice-based two-round multi-signature with trapdoor-free simulation. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part V*, volume 14085 of *LNCS*, pages 716–747. Springer, Cham, August 2023.

CKGW22. Deirdre Connolly, Chelsea Komlo, Ian Goldberg, and Christopher A. Wood. Two-Round Threshold Schnorr Signatures with FROST. Internet-Draft draft-irtf-cfrg-frost-10, Internet Engineering Task Force, September 2022. Work in Progress.

CKM23a. Elizabeth C. Crites, Chelsea Komlo, and Mary Maller. Fully adaptive Schnorr threshold signatures. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part I*, volume 14081 of *LNCS*, pages 678–709. Springer, Cham, August 2023.

CKM+23b. Elizabeth C. Crites, Chelsea Komlo, Mary Maller, Stefano Tessaro, and Chenzhi Zhu. Snowblind: A threshold blind signature in pairing-free groups. In Helena Handschuh and Anna Lysyanskaya, editors, *CRYPTO 2023, Part I*, volume 14081 of *LNCS*, pages 710–742. Springer, Cham, August 2023.

CS19. Daniele Cozzo and Nigel P. Smart. Sharing the LUOV: Threshold post-quantum signatures. In Martin Albrecht, editor, *17th IMA International Conference on Cryptography and Coding*, volume 11929 of *LNCS*, pages 128–153. Springer, Cham, December 2019.

DDFY94. Alfredo De Santis, Yvo Desmedt, Yair Frankel, and Moti Yung. How to share a function securely. In *26th ACM STOC*, pages 522–533. ACM Press, May 1994.

Des88. Yvo Desmedt. Society and group oriented cryptography: A new concept. In Carl Pomerance, editor, *CRYPTO'87*, volume 293 of *LNCS*, pages 120–127. Springer, Berlin, Heidelberg, August 1988.

DF90. Yvo Desmedt and Yair Frankel. Threshold cryptosystems. In Gilles Brassard, editor, *CRYPTO'89*, volume 435 of *LNCS*, pages 307–315. Springer, New York, August 1990.

DK01. Ivan Damgård and Maciej Koprowski. Practical threshold RSA signatures without a trusted dealer. In Birgit Pfitzmann, editor, *EUROCRYPT 2001*, volume 2045 of *LNCS*, pages 152–165. Springer, Berlin, Heidelberg, May 2001.

DKL+18. Léo Ducas, Eike Kiltz, Tancrède Lepoint, Vadim Lyubashevsky, Peter Schwabe, Gregor Seiler, and Damien Stehlé. CRYSTALS-Dilithium: A lattice-based digital signature scheme. *IACR TCHES*, 2018(1):238–268, 2018.

DOTT21. Ivan Damgård, Claudio Orlandi, Akira Takahashi, and Mehdi Tibouchi. Two-round n-out-of-n and multi-signatures and trapdoor commitment from lattices. In Juan Garay, editor, *PKC 2021, Part I*, volume 12710 of *LNCS*, pages 99–130. Springer, Cham, May 2021.

DT06. Ivan Damgård and Rune Thorbek. Linear integer secret sharing and distributed exponentiation. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malkin, editors, *PKC 2006*, volume 3958 of *LNCS*, pages 75–90. Springer, Berlin, Heidelberg, April 2006.

EKT24. Thomas Espitau, Shuichi Katsumata, and Kaoru Takemure. Two-round threshold signature from algebraic one-more learning with errors. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 387–424. Springer, Cham, August 2024.

ENP24. Thomas Espitau, Guilhem Niot, and Thomas Prest. Flood and submerse: Distributed key generation and robust threshold signature from lattices. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 425–458. Springer, Cham, August 2024.

GG18. Rosario Gennaro and Steven Goldfeder. Fast multiparty threshold ECDSA with fast trustless setup. In David Lie, Mohammad Mannan, Michael Backes, and XiaoFeng Wang, editors, *ACM CCS 2018*, pages 1179–1194. ACM Press, October 2018.

GGN16. Rosario Gennaro, Steven Goldfeder, and Arvind Narayanan. Threshold-optimal DSA/ECDSA signatures and an application to bitcoin wallet security. In Mark Manulis, Ahmad-Reza Sadeghi, and Steve Schneider, editors, *ACNS 16International Conference on Applied Cryptography and Network Security*, volume 9696 of *LNCS*, pages 156–174. Springer, Cham, June 2016.

GJKR96. Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Robust threshold DSS signatures. In Ueli M. Maurer, editor, *EUROCRYPT'96*, volume 1070 of *LNCS*, pages 354–371. Springer, Berlin, Heidelberg, May 1996.

GJKR03. Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Secure applications of Pedersen's distributed key generation protocol. In Marc Joye, editor, *CT-RSA 2003*, volume 2612 of *LNCS*, pages 373–390. Springer, Berlin, Heidelberg, April 2003.

GJKR07. Rosario Gennaro, Stanislaw Jarecki, Hugo Krawczyk, and Tal Rabin. Secure distributed key generation for discrete-log based cryptosystems. *Journal of Cryptology*, 20(1):51–83, January 2007.

GKS24. Kamil Doruk Gür, Jonathan Katz, and Tjerand Silde. Two-round threshold lattice-based signatures from threshold homomorphic encryption. In Markku-Juhani Saarinen and Daniel Smith-Tone, editors, *Post-Quantum Cryptography - 15th International Workshop, PQCrypto 2024, Part II*, pages 266–300. Springer, Cham, June 2024.

GMPW20. Nicholas Genise, Daniele Micciancio, Chris Peikert, and Michael Walter. Improved discrete gaussian and subgaussian analysis for lattice cryptography. In Aggelos Kiayias, Markulf Kohlweiss, Petros Wallden, and Vassilis Zikas, editors, *PKC 2020, Part I*, volume 12110 of *LNCS*, pages 623–651. Springer, Cham, May 2020.

GPV08. Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008.

GRJK00. Rosario Gennaro, Tal Rabin, Stanislaw Jarecki, and Hugo Krawczyk. Robust and efficient sharing of RSA functions. *Journal of Cryptology*, 13(2):273–300, March 2000.

HKLN20. Eduard Hauck, Eike Kiltz, Julian Loss, and Ngoc Khanh Nguyen. Lattice-based blind signatures, revisited. In Daniele Micciancio and Thomas Ristenpart, editors, *CRYPTO 2020, Part II*, volume 12171 of *LNCS*, pages 500–529. Springer, Cham, August 2020.

HMP06. Shlomo Hoory, Avner Magen, and Toniann Pitassi. Monotone circuits for the majority function. In *International Workshop on Approximation Algorithms for Combinatorial Optimization*, pages 410–425. Springer, 2006.

KG20. Chelsea Komlo and Ian Goldberg. FROST: Flexible round-optimized Schnorr threshold signatures. In Orr Dunkelman, Michael J. Jacobson Jr., and Colin O'Flynn, editors, *SAC 2020*, volume 12804 of *LNCS*, pages 34–65. Springer, Cham, October 2020.

KLS18. Eike Kiltz, Vadim Lyubashevsky, and Christian Schaffner. A concrete treatment of Fiat-Shamir signatures in the quantum random-oracle model. In Jesper Buus Nielsen and Vincent Rijmen, editors, *EUROCRYPT 2018, Part III*, volume 10822 of *LNCS*, pages 552–586. Springer, Cham, April / May 2018.

KRT24. Shuichi Katsumata, Michael Reichle, and Kaoru Takemure. Adaptively secure 5 round threshold signatures from MLWE/MSIS and DL with rewinding. In Leonid Reyzin and Douglas Stebila, editors, *CRYPTO 2024, Part VII*, volume 14926 of *LNCS*, pages 459–491. Springer, Cham, August 2024.

KW93. M. Karchmer and A. Wigderson. On span programs. In *[1993] Proceedings of the Eigth Annual Structure in Complexity Theory Conference*, pages 102–111, 1993.

LDK+22. Vadim Lyubashevsky, Léo Ducas, Eike Kiltz, Tancrède Lepoint, Peter Schwabe, Gregor Seiler, Damien Stehlé, and Shi Bai. CRYSTALS-DILITHIUM. Technical report, National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022.

Lin24. Yehuda Lindell. Simple three-round multiparty schnorr signing with full simulatability. *CiC*, 1(1):25, 2024.

LNR18. Yehuda Lindell, Ariel Nof, and Samuel Ranellucci. Fast secure multiparty ECDSA with practical distributed key generation and applications to cryptocurrency custody. Cryptology ePrint Archive, Report 2018/987, 2018.

Lyu09. Vadim Lyubashevsky. Fiat-Shamir with aborts: Applications to lattice and factoring-based signatures. In Mitsuru Matsui, editor, *ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 598–616. Springer, Berlin, Heidelberg, December 2009.

Lyu12. Vadim Lyubashevsky. Lattice signatures without trapdoors. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 738–755. Springer, Berlin, Heidelberg, April 2012.

MP13. Daniele Micciancio and Chris Peikert. Hardness of SIS and LWE with small parameters. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 21–39. Springer, Berlin, Heidelberg, August 2013.

MR09. Daniele Micciancio and Oded Regev. Lattice-based cryptography. In *Post-quantum cryptography*, pages 147–191. Springer, 2009.

Natnt. National Institute of Standards and Technology. Multi-Party Threshold Cryptography, 2018–Present. https://csrc.nist.gov/Projects/threshold-cryptography.

NRS21. Jonas Nick, Tim Ruffing, and Yannick Seurin. MuSig2: Simple two-round Schnorr multi-signatures. In Tal Malkin and Chris Peikert, editors, *CRYPTO 2021, Part I*, volume 12825 of *LNCS*, pages 189–221, Virtual Event, August 2021. Springer, Cham.

PFH+22. Thomas Prest, Pierre-Alain Fouque, Jeffrey Hoffstein, Paul Kirchner, Vadim Lyubashevsky, Thomas Pornin, Thomas Ricosset, Gregor Seiler, William Whyte, and Zhenfei Zhang. FALCON. Technical report, National Institute of Standards and Technology, 2022. available at https://csrc.nist.gov/Projects/post-quantum-cryptography/selected-algorithms-2022.

PKM+24. Rafaël Del Pino, Shuichi Katsumata, Mary Maller, Fabrice Mouhartem, Thomas Prest, and Markku-Juhani O. Saarinen. Threshold raccoon: Practical threshold signatures from standard lattice assumptions. In Marc Joye and Gregor Leander, editors, *EUROCRYPT 2024, Part II*, volume 14652 of *LNCS*, pages 219–248. Springer, Cham, May 2024.

Rén61. Alfréd Rényi. On measures of entropy and information. In *Proceedings of the fourth Berkeley symposium on mathematical statistics and probability, volume 1: contributions to the theory of statistics*, volume 4, pages 547–562. University of California Press, 1961.

Ros20. Mélissa Rossi. *Extended security of lattice-based cryptography.* PhD thesis, Université Paris sciences et lettres, 2020.

RS62. J Barkley Rosser and Lowell Schoenfeld. Approximate formulas for some functions of prime numbers. *Illinois Journal of Mathematics*, 6(1):64–94, 1962.

Sho00. Victor Shoup. Practical threshold signatures. In Bart Preneel, editor, *EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 207–220. Springer, Berlin, Heidelberg, May 2000.

SS01. Douglas R. Stinson and Reto Strobl. Provably secure distributed Schnorr signatures and a (t, n) threshold scheme for implicit certificates. In Vijay Varadharajan and Yi Mu, editors, *ACISP 01*, volume 2119 of *LNCS*, pages 417–434. Springer, Berlin, Heidelberg, July 2001.

TT15. Katsuyuki Takashima and Atsushi Takayasu. Tighter security for efficient lattice cryptography via the Rényi divergence of optimized orders. In Man Ho Au and Atsuko Miyaji, editors, *ProvSec 2015*, volume 9451 of *LNCS*, pages 412–431. Springer, Cham, November 2015.

TZ23. Stefano Tessaro and Chenzhi Zhu. Threshold and multi-signature schemes from linear hash functions. In Carmit Hazay and Martijn Stam, editors, *EUROCRYPT 2023, Part V*, volume 14008 of *LNCS*, pages 628–658. Springer, Cham, April 2023.

Val84. L.G Valiant. Short monotone formulae for the majority function. *Journal of Algorithms*, 5(3):363–366, 1984.

A Proof of Lemma 10

We first state the two following lemmas borrowed from [GMPW20] and used in our proof. The latter one is adapted to our Gaussian notation, which we give a proof for completeness.

Lemma 21 (Corollary 2.7 of [GMPW20]). *For any lattice $\Lambda \subseteq \mathbb{R}^m$ and $\varepsilon \in (0,1)$ where $\eta_\varepsilon(\Lambda) \leqslant 1$, we have that for any $\boldsymbol{x} \in \mathbb{R}^m$,*

$$\rho(\Lambda + \boldsymbol{x}) \in [1-\varepsilon, 1+\varepsilon]\frac{\rho(\boldsymbol{x}_{\perp\Lambda})}{\Delta(\Lambda)}$$

where $\boldsymbol{x}_{\perp\Lambda}$ is the projection of $\boldsymbol{x}$ orthogonal to Λ and $\Delta(\Lambda)$ is the determinant of the lattice Λ defined as the volume of its fundamental parallelepiped $\mathcal{P}(\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}) := \{\sum_{i=1}^m x_i\boldsymbol{b}_i : \forall i \in [k], x_i \in [0,1)\}$ for any basis $\{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}$ of Λ.

Lemma 22 (Lemma 2.3 of [GMPW20] adapted to our notations). *For any lattice coset $A = \Lambda + \boldsymbol{a} \subseteq \mathbb{R}^m$ and any full-row-rank $T \in \mathbb{R}^{k\times m}$ such that T is injective on A, we have that the distributions $T \cdot \mathscr{D}_A^m$ and $\mathscr{D}_{TA,TT^T}^k$ are identical.*

Proof. First, because T is injective on A, for each $\boldsymbol{y} \in TA$, one can write it as $T\boldsymbol{x}$ for a unique $\boldsymbol{x} \in A$. Hence, it suffices to show that for any $\boldsymbol{y} = T\boldsymbol{x} \in TA$, $\rho_{TT^T}(\boldsymbol{y}) = \rho(\boldsymbol{x})$ to conclude the proof. Then, consider

$$\rho_{TT^T}(\boldsymbol{y}) = \exp(-\pi \boldsymbol{y}^T (TT^T)^{-1}\boldsymbol{y}) = \exp(-\pi \boldsymbol{x}^T T^T (TT^T)^{-1} T\boldsymbol{x}) \; .$$

Next, we will show that $\boldsymbol{x}^T T^T (TT^T)^{-1} T\boldsymbol{x} = \|\boldsymbol{x}\|^2$ for any $\boldsymbol{x} \in A$. By singular value decomposition, we can write T as $T = UDV^T$ for orthonormal matrices $U \in \mathbb{R}^{k\times k}, V \in \mathbb{R}^{m\times m}$ (i.e., $UU^T = \mathbb{I}_k, VV^T = \mathbb{I}_m$) and a rectangular diagonal matrix $D \in \mathbb{R}^{k\times m}$. Also, with T being full-row-rank, $D_{11}, \ldots, D_{kk} \neq 0$. This means that the first k columns of V span a subspace of $\mathbb{R}^m$ on which T is injective (we can see this by considering $T\boldsymbol{v}_i = D_{ii}\boldsymbol{u}_i$ for each column vector $\boldsymbol{v}_i$ of V for $i \in [k]$, so $\{T\boldsymbol{v}_i\}_{i\in[k]}$ spans $\mathbb{R}^k$). Therefore, $\boldsymbol{x} \in A$ can be written as $\sum_{i=1}^k c_i \boldsymbol{v}_i$ and $\|x\|^2 = \sum_{i=1}^k c_i^2$. Then, see that $(TT^T)^{-1} = U(DD^T)^{-1}U^T$, so $T^T(TT^T)^{-1}T = VI'V^T$ where $I' \in \mathbb{R}^{m\times m}$ is a diagonal matrix with 1 in its first k diagonal entries and 0 otherwise. Finally, we conclude that $\boldsymbol{x}^T VI'V^T \boldsymbol{x} = \sum_{j=1}^k c_j^2 = \boldsymbol{x}^T\boldsymbol{x}$, proving the lemma. □

Proof (of Lemma 10). We note first that since $\phi_{\mathsf{M}}(T)$ has full-row-rank, for any $\boldsymbol{x} \in K_{\mathbb{R}}^m \backslash \{0\}$,

$$\left\langle \boldsymbol{x}, TT^\dagger \boldsymbol{x} \right\rangle = \phi(\boldsymbol{x})^T \phi_M(TT^\dagger)\phi(\boldsymbol{x}) = \phi(\boldsymbol{x}^T)\phi_M(T^\dagger)^T\phi_M(T^\dagger)\phi(\boldsymbol{x}) = \left\|\phi_{\mathsf{M}}(T^\dagger)\phi(\boldsymbol{x})\right\|_2^2 \neq 0 \; .$$

Hence, $\Sigma = TT^\dagger$ is positive definite. Since $T \in K^{k\times m}$, the determinant $\Delta(\Sigma) \in K\backslash\{0\}$ and thus Σ is invertible. Now, denote $T' = \phi_{\mathsf{M}}(T) \in \mathbb{R}^{kN\times mN}$. We know that $\ker(T') = \phi(\ker(T))$ since for any $\boldsymbol{x} \in K_{\mathbb{R}}^m$ that $T\boldsymbol{x} = \boldsymbol{0}$, $\phi_{\mathsf{M}}(T)\phi(\boldsymbol{x}) = \boldsymbol{0}$, and similarly for any $\boldsymbol{x} \in \mathbb{R}^{mN}$ that $T'\boldsymbol{x} = \boldsymbol{0}$, $T\phi^{-1}(\boldsymbol{x}) = \boldsymbol{0}$. Additionally, because $\ker(T)$ is a Λ-subspace, (i.e., $\mathsf{Span}(\Lambda \cap \ker(T)) = \ker(T)$), we have that $\ker(T') = \phi(\ker(T)) = \phi(\mathsf{Span}(\Lambda \cap \ker(T))) = \mathsf{Span}(\phi(\Lambda) \cap \phi(\ker(T))) = \mathsf{Span}(\phi(\Lambda) \cap \ker(T'))$, so $\ker(T')$ is $\phi(\Lambda)$-subspace.

Then, we consider the coefficient embedding of the two distributions, which by Lemma 2, we can see that the coefficient embedding of values from $T \cdot \mathscr{D}^m_{\Lambda+\boldsymbol{a},\sigma}$ and $\mathscr{D}^k_{T\Lambda+T\boldsymbol{a},\sigma^2TT^\dagger}$ have the same distribution as $T' \cdot \mathscr{D}^{mN}_{\phi(\Lambda+\boldsymbol{a}),\sigma}$ and $\mathscr{D}^{kN}_{\phi(T\Lambda+T\boldsymbol{a}),\sigma^2\phi_{\mathsf{M}}(TT^\dagger)}$, respectively. We note that $\phi(T\Lambda + T\boldsymbol{a}) = T'\phi(\Lambda + \boldsymbol{a})$ and $\phi_{\mathsf{M}}(TT^\dagger) = T'T'^T$. Additionally, denote $\Lambda' = \frac{1}{\sigma}\phi(\Lambda) \subseteq \mathbb{R}^{mN}$, $\boldsymbol{a}' = \frac{1}{\sigma}\phi(\boldsymbol{a}) \in \mathbb{R}^{mN}$ and $A' = \Lambda' + \boldsymbol{a}'$. Then, $T' \cdot \mathscr{D}^{mN}_{\phi(\Lambda+\boldsymbol{a}),\sigma} = \sigma T' \cdot \mathscr{D}^{mN}_{A'}$ and $\mathscr{D}^{kN}_{T'\phi(\Lambda+\boldsymbol{a}),\sigma^2\phi_{\mathsf{M}}(TT^\dagger)} = \sigma \cdot \mathscr{D}^{kN}_{T'A',T'T'^T}$. Note that $\eta_\varepsilon(\Lambda') \leqslant 1$ since $\eta_\varepsilon(\Lambda) \leqslant \sigma$. Thus, our goal now is to show that

$$\sigma T' \cdot \mathscr{D}^{mN}_{A'} \overset{\varepsilon'}{\approx} \sigma \cdot \mathscr{D}^{kN}_{T'A',T'T'^T} \; .$$

To do this, let $P = \ker(T')$ and consider the projection $\boldsymbol{x}_{\perp P}$ of any $\boldsymbol{x} \in \mathbb{R}^{mN}$ orthogonal to P. Observe that for any $\boldsymbol{x}$, $T'\boldsymbol{x} = T'\boldsymbol{x}_{\perp P}$. Then, for a distribution $(\mathscr{D}^{mN}_{A'})_{\perp P}$ of $\boldsymbol{x}_{\perp P}$ with $\boldsymbol{x}$ sampled from $\mathscr{D}^{mN}_{A'}$, we have that $\sigma T' \cdot (\mathscr{D}^{mN}_{A'})_{\perp P}$ is identically distributed to $\sigma T' \cdot \mathscr{D}^{mN}_{A'}$. Also, consider a lattice coset $A'_{\perp P}$ which is obtained by projecting each vector in A' orthogonally to P (this is a well-defined lattice coset, because P is a Λ'-subspace). Also, since T' is injective on $A'_{\perp P}$, by Lemma 22, $T' \cdot \mathscr{D}^{mN}_{A'_{\perp P}}$ and $\mathscr{D}^{kN}_{T'A'_{\perp P},T'T'^T}$ are identically distributed. Hence, we only need to show that $\sigma T' \cdot (\mathscr{D}^{mN}_{A'})_{\perp P} \overset{\varepsilon'}{\approx} \sigma T' \cdot \mathscr{D}^{mN}_{A'_{\perp P}}$, which can be done by applying the data processing property of R_∞ (Lemma 6) and showing that

$$(\mathscr{D}^{mN}_{A'})_{\perp P} \overset{\varepsilon'}{\approx} \mathscr{D}^{mN}_{A'_{\perp P}} \; .$$

To show this final step, first see that both distributions have the same support $A'_{\perp P}$. Then, for each $\boldsymbol{x} \in A'_{\perp P}$, consider the probability that we sample $\boldsymbol{x}$ from $(\mathscr{D}^{mN}_{A'})_{\perp P}$ and $\mathscr{D}^{mN}_{A'_{\perp P}}$ respectively. Also, let $\varLambda_P = \varLambda' \cap P$ and $\boldsymbol{w}' \in A'$ be some vector where $\boldsymbol{w}'_{\perp P} = \boldsymbol{x}$. Then,

$$(\mathscr{D}^{mN}_{A'})_{\perp P}(\boldsymbol{x}) = \frac{\rho(\{\boldsymbol{w} \in A' : \boldsymbol{w}_{\perp P} = \boldsymbol{x}\})}{\rho(A')} = \frac{\rho(\boldsymbol{w}' + \varLambda_P)}{\rho(A')} \in [1-\varepsilon, 1+\varepsilon] \cdot \frac{\rho(\boldsymbol{w}'_{\perp P})}{\rho(A')\Delta(\varLambda_P)} ,$$

where the last step follows from Lemma 21 and that $\eta_\varepsilon(\varLambda') \leqslant 1$. Next, $\mathscr{D}^{mN}_{A'_{\perp P}}(\boldsymbol{x}) = \rho(\boldsymbol{x})/\rho(A'_{\perp P})$, so we can write $\frac{\rho(\boldsymbol{w}'_{\perp P})}{\rho(A')\Delta(\varLambda_P)}$ as $C \cdot \mathscr{D}^{mN}_{A'_{\perp P}}(\boldsymbol{x})$ where $C = \frac{\rho(A'_{\perp P})}{\rho(A')\Delta(\varLambda_P)}$. By summing over all $\boldsymbol{x} \in A'_{\perp P}$, we have that $C \in [\frac{1}{1+\varepsilon}, \frac{1}{1-\varepsilon}]$. Thus, $(\mathscr{D}^{mN}_{A'})_{\perp P}(\boldsymbol{x}) \in [\frac{1-\varepsilon}{1+\varepsilon}, \frac{1+\varepsilon}{1-\varepsilon}] \cdot \mathscr{D}^{mN}_{A'_{\perp P}}(\boldsymbol{x})$ for any $\boldsymbol{x} \in A'_{\perp P}$, implying $(\mathscr{D}^{mN}_{A'})_{\perp P} \overset{\varepsilon'}{\approx} \mathscr{D}^{mN}_{A'_{\perp P}}$. □

B Proof of Lemma 18

Proof. First, let H_i denote the distribution where h_i is sampled from according to HG. One can view HG as independently sampling $h_i \leftarrow H_i$ for $i \in [q]$. For any $i \in S$, $h_1, \ldots, h_{i-1}$, and input x, define

$$Y_i(x, h_1, \ldots, h_{i-1}) := \Pr_{h_i \leftarrow H_i, \ldots, h_q \leftarrow H_q}[I = i : (I, \mathrm{Out}) \leftarrow \mathcal{A}(x, h_1, \ldots, h_q)] .$$

Then, we have

$$\begin{aligned}\mathrm{acc}(\mathcal{A}) &= \sum_{i \in S} \Pr_{\substack{x \leftarrow \mathsf{IG}, \\ (h_1,\ldots,h_q) \leftarrow \mathsf{HG}}} [I = i : (I, \mathrm{Out}) \leftarrow \mathcal{A}(x, h_1, \ldots, h_q)] \\ &= \sum_{i \in S} \mathop{\mathbb{E}}_{\substack{x \leftarrow \mathsf{IG}, \\ h_1 \leftarrow H_1, \ldots, h_{i-1} \leftarrow H_{i-1}}} [Y_i(x, h_1, \ldots, h_{i-1})] .\end{aligned}$$

Thus, we have

$$\begin{aligned}\mathrm{acc}(\mathsf{Fork}^{\mathcal{A}}) &= \sum_{i \in S} \Pr_{\substack{x \leftarrow \mathsf{IG}, (h_1,\ldots,h_q) \leftarrow \mathsf{HG}, \\ \bar{h}_i \leftarrow H_i, \ldots, \bar{h}_q \leftarrow H_q}} \left[I = \bar{I} = i : \begin{array}{c} (I, \mathrm{Out}) \leftarrow \mathcal{A}(x, h_1, \ldots, h_q), \\ (\bar{I}, \overline{\mathrm{Out}}) \leftarrow \mathcal{A}(x, h_1, \ldots, h_{i-1}, \bar{h}_i, \ldots, \bar{h}_q) \end{array} \right] \\ &= \sum_{i \in S} \mathop{\mathbb{E}}_{\substack{x \leftarrow \mathsf{IG}, \\ h_1 \leftarrow H_1, \ldots, h_{i-1} \leftarrow H_{i-1}}} [Y_i(x, h_1, \ldots, h_{i-1})^2] \\ &\geqslant \sum_{i \in S} \left(\mathop{\mathbb{E}}_{\substack{x \leftarrow \mathsf{IG}, \\ h_1 \leftarrow H_1, \ldots, h_{i-1} \leftarrow H_{i-1}}} [Y_i(x, h_1, \ldots, h_{i-1})] \right)^2 \\ &\geqslant \frac{1}{|S|} \cdot \left(\sum_{i \in S} \mathop{\mathbb{E}}_{\substack{x \leftarrow \mathsf{IG}, \\ h_1 \leftarrow H_1, \ldots, h_{i-1} \leftarrow H_{i-1}}} [Y_i(x, h_1, \ldots, h_{i-1})] \right)^2 \\ &= \frac{\mathrm{acc}(\mathcal{A})^2}{|S|} ,\end{aligned}$$

where the first inequality is due to the fact that $\mathsf{E}[X^2] \geqslant (\mathsf{E}[X])^2$ and the second inequality is due to the fact that $\sum_{i=1}^n a_i^2 \geqslant \frac{1}{n} \left(\sum_{i=1}^n a_i\right)^2$. □