Research Article

Lattice-Based Linearly Homomorphic Signature Scheme over $\mathbb{F}_2$

Jie Cai ¹, Han Jiang ², Hao Wang,³ and Qiuliang Xu²

¹School of Science, Shandong Jianzhu University, Ji'nan, Shandong, China

²School of Software, Shandong University, Ji'nan, Shandong, China

³School of Information Science and Engineering, Shandong Normal University, Ji'nan, Shandong, China

Correspondence should be addressed to Han Jiang; jianghan@sdu.edu.cn

Received 7 August 2020; Revised 2 September 2020; Accepted 22 September 2020; Published 29 October 2020

Academic Editor: Zhihua Xia

Copyright © 2020 Jie Cai et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, we design a new lattice-based linearly homomorphic signature scheme over $\mathbb{F}_2$. The existing schemes are all constructed based on hash-and-sign lattice-based signature framework, where the implementation of preimage sampling function is Gaussian sampling, and the use of trapdoor basis needs a larger dimension ($m \geq 5n \log q$). Hence, they cannot resist potential side-channel attacks and have larger sizes of public key and signature. Under Fiat-Shamir with aborting signature framework and general SIS problem restricted condition ($m \geq n \log q$), we use uniform sampling of filtering technology to design the scheme, and then, our scheme has a smaller public key size and signature size than the existing schemes and it can resist side-channel attacks.

1. Introduction

The idea of the linear homomorphic signature scheme comes from network coding routing mechanism. Specifically, after a signer sends a number of signatures for messages to router (verifier) in a computer network using network coding, the router can generate a random linear combination μ of the received messages. Using the homomorphic property, the router computes a signature (σ) of (μ) and transmits (σ, μ) to the next router, and the process will be continued for different linear combined messages. The final router accepts properly signed signature and recovers the original message by solving a full-rank linear system over $\mathbb{F}_p$.

Then, one can easily abstract definition from applications. Informally, given n -dimensional message vectors ($\mu_1, \dots, \mu_k \in \mathbb{F}_p^n$) and signatures $\sigma_1, \dots, \sigma_k$, anyone can create a signature for any vector $\mu \in \text{span}\{\mu_1, \dots, \mu_k\}$. At the same time, if any adversary cannot produce a valid signature for $\mu' \notin \text{span}\{\mu_1, \dots, \mu_k\}$, we say the linear homomorphic signature scheme is secure. There exist many classical linearly homomorphic signatures [1–4] based on the difficulty in solving discrete logarithm or the difficulty in integer factoring. However, they have two obvious disadvantages.

First, the parameter p must be large enough to guarantee the difficulties in classical problems, but implementations are generally given over $\mathbb{F}_2$ in network coding. Second, these schemes cannot resist quantum computing attack as we all know. Hence, more and more people focus on designing postquantum linearly homomorphic signature scheme over $\mathbb{F}_2$, where lattice-based schemes are significant.

2. Related Work

2.1. Lattice-Based Signature Schemes. The existing lattice-based signature schemes are mostly based on short integer solution (SIS) problem first provided in [5] ($\mathbf{x}: \mathbf{A}\mathbf{s} = 0 \bmod q, \|\mathbf{s}\| \leq \beta, \mathbf{A} \in \mathbb{Z}_q^{n \times m}, \mathbf{s} \in \mathbb{Z}^m$). There are two frameworks to construct lattice-based signature schemes: hash-and-sign type [6, 7] and Fiat-Shamir type [8–11]. In schemes of hash-and-sign type, the signer uses trapdoor basis to compute preimage sampling function to create signature (σ) satisfying $\mathbf{A}\sigma = H(\mu) \bmod q$, where H is a hash function. Unlike the hash-and-sign lattice-based signature framework, aborting technology is used in schemes of Fiat-Shamir type without trapdoor. This is the output of the signature ($\mathbf{z} = \mathbf{s}\mathbf{c} + \mathbf{y}$) according to some probabilities (rejection sampling) or the norm of signature must be in a security range (filtering outputting), where

($\mathbf{c} = H(\mathbf{A}\mathbf{y} \bmod q, \mu)$) and ($\mathbf{y} \leftarrow \mathcal{D}_y$) ($\mathcal{D}$ is a Gaussian distribution or uniform distribution).

Furthermore, Gaussian distribution is utilized in preimage sampling and rejection sampling, which cannot resist partial side-channel attacks (see [12, 13]). Although, the author in [14] showed that an almost perfect implementation could resist these attacks, and the designed programme errors might occur.

2.2. Lattice-Based Linearly Homomorphic Signature. The first lattice-based linearly homomorphic signature scheme over $\mathbb{F}_2$ was proposed by Boneh and Freeman [15] in 2011, which was based on k -SIS problem, that is, finding a solution $\mathbf{s}$ satisfying $\mathbf{s} \notin \text{span}\{\mathbf{s}_1, \dots, \mathbf{s}_k\}$ under giving ($\mathbf{A}\mathbf{s}_i = 0 \bmod q$). In addition, the specific sign process is ($\mathbf{A}\sigma = H(\mu) \bmod 2q$), where ($\mathbf{A} \in \mathbb{Z}_{2q}^{n \times m}$). Soon after, Wang et al. proposed an improved scheme [16] based on the general SIS problem, and the size of public key and signature is smaller than [15] by changing $2q$ into q . Compared to signature size, ($2m + 2m \log q + n$) in scheme [15, 16] has the smaller size ($m \log q + n$).

In fact, both of them are designed in terms of the hash-and-sign lattice-based signature framework [6], where Gaussian sampling is used inevitably. Meanwhile, the generation of trapdoor basis needs that lattice dimension is ($m \geq 5n \log q$) (see [17, 18]), which is larger than ($m \geq n \log q$) for SIS problem itself.

2.3. Our Contributions. In this paper, our scheme overcomes the drawbacks of existing schemes. Specifically, based on the SIS problem, we use filtering technology of Fiat-Shamir with aborting signature framework to design a new linearly homomorphic signature scheme over $\mathbb{F}_2$, and the advantages can be seen as follows:

- (1) The signature size is smaller than existing lattice-based schemes. The signature of our scheme is $\sigma = (\mathbf{z}, \mathbf{h}, \tau)$, and the size is ($2m \log q + n$), where ($m \geq n \log q$). Since our design does not utilize preimage trapdoor sampling, the signature size is smaller than ($m' \log q + n$) in [16] with the same n and q , where ($m' \geq 5n \log q$). Here, we use a different lattice dimension m' to distinguish the difference in signature size.
- (2) Our scheme can resist side-channel attacks. Using filtering technology, the masked element $\mathbf{y}$ is chosen uniformly at random under restriction ($\|\mathbf{y}\|_\infty \leq \gamma$), and $\mathbf{z}$ must satisfy the condition $\|\mathbf{z}\|_\infty \leq \gamma - \beta$; otherwise, $\mathbf{z} \leftarrow \perp$ (aborting). Hence, the signature output can protect secret key, and the scheme can resist side-channel attacks without Gaussian sampling.

2.4. Organization of the Paper. We will provide two main technical descriptions to show how our scheme can have the above advantages in Section 3. Then, we propose the basic notations and definitions of linearly homomorphic signature in

Section 4. We show the detailed design and security proof of our lattice-based linearly homomorphic signature in Section 5 and Section 6, respectively. In Section 7, we present efficiency comparisons. Finally, we give a conclusion and further work in Section 8. Data availability, conflicts of interest, and funding statement can be seen in the last three sections, respectively.

3. Technical Notes

In this part, we give detailed descriptions to show how we get a smaller signature size and the scheme can resist side-channel attacks.

3.1. Different Lattice Dimension Assumptions. As we know, given security parameter n, m influences the signature size directly. Thus, we want to reduce it. Fortunately, compared to hash-and-sign signature framework, the Fiat-Shamir signature framework has advantage in this aspect. We show the main reason as follows.

Definition 1. (the short integer solution problem SIS). Given m uniformly random elements ($\mathbf{a}_i \in \mathbb{Z}_q^n$), find a nonzero ($\mathbf{z} \in \mathbb{Z}^m$) of norm ($\|\mathbf{z}\| \leq \beta$) satisfying

$$\sum_{i=1}^m \mathbf{a}_i \mathbf{z}_i = 0 \in \mathbb{Z}_q^n. \quad (1)$$

Usually, it is denoted $\mathbf{A}\mathbf{z} = 0 \in \mathbb{Z}_q^n$, where $\mathbf{a}_i \in \mathbb{Z}_q^n$ forms the columns of $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$. To guarantee the hardness (existence of solution) of this problem, the parameters satisfy conditions $\sqrt{m} \leq \beta < q, m \geq n \log q, q \in \mathbb{Z}^+, n \geq 100$. Normally, people consider the inhomogeneous version of the SIS problem, which is to find a small solution of equation $\mathbf{A}\mathbf{z} = \mathbf{b}$.

To design Fiat-Shamir signature schemes, the lattice dimension m satisfies $m \geq n \log q$ enough. However, hash-and-sign type needs $m \geq 5n \log q$ to get trapdoor basis; thus, we provide the existing conclusion below.

Proposition 1 (see [6, 17, 18]). *Given any prime q and ($m \geq 5n \log q$), then there exists a PPT algorithm which outputs ($\mathbf{A} \in \mathbb{Z}_q^{n \times m}$) statistically close to uniform over ($\mathbb{Z}_q^{n \times m}$) and a full-rank set ($\mathbf{S} \subset \Lambda^\perp(\mathbf{A}), \|\mathbf{S}\| \leq m^{2.5}$) by input 1^n . Then, it further gets a good basis ($\mathbf{T} \subset \Lambda^\perp(\mathbf{A})$) satisfying ($\|\mathbf{T}\| \leq \|\mathbf{S}\|$).*

From what has been discussed above, we get the smaller m the better for the size of signature and secret key under the same n . Hence, we design a new scheme using Fiat-Shamir signature framework without trapdoor (basis).

3.2. Filtering Technology. Since the existing lattice-based linearly homomorphic signature schemes are based on the hash-and-sign signature framework in which they use preimage sampling function implemented by Gaussian sampling, the schemes cannot resist side-channel attacks. Hence, we unitize uniform sampling of Fiat-Shamir framework to generate signature.

The idea of filtering can be traced back to [10, 19], and it is formally provided in [20] to design a blind signature scheme. Here, we rewrite this lemma according to our construction.

Lemma 1. For arbitrary $\mathbf{a} \in \{\mathbf{x} \in \mathbb{Z}^k: \|\mathbf{x}\|_\infty \leq \beta\}$ and random $(\mathbf{b} \leftarrow \{\mathbf{x} \in \mathbb{Z}^k: \|\mathbf{x}\|_\infty \leq \gamma\}, k = \Omega(n))$, if $(\gamma \geq \phi k \beta)$ with $(\phi \in \mathbb{N}^+)$, then we have $(\Pr[\|\mathbf{a} - \mathbf{b}\|_\infty \leq \gamma - \beta] > (1/((e^{(1/\phi)} - o(1))))$.

Then, the repeat time of our scheme can be computed by $e^{(1/\phi)}$. Intuitively, bigger ϕ size is better. However, this value has influence on the size of $(\|\mathbf{a} - \mathbf{b}\|_\infty)$ directly; thus, it leads to increased communication costs. Hence, we can assign this value according to different efficiency requirements, which is a nice advantage in practice. According to [20], the authors provide the condition $(\phi = 4)$ is the best; then, the repeat time is no more than 2, which is also hold for our scheme.

In our scheme, the parameters (γ, β) satisfy $(\gamma, \beta < q)$. In addition, to facilitate comparisons with schemes [15, 16], we use $(\|\mathbf{a} - \mathbf{b}\|_\infty \leq q)$ to compute signature size instead of $(\gamma - \beta)$ (see Table 1).

4. Preliminaries

4.1. Notions. We denote $(\mathcal{R} = \mathbb{Z})$. The elements in $\mathcal{R}$ (vector or matrix) are marked in bold, $\|\mathbf{y}\|$ is l_2 norm, and $\|\mathbf{y}\|_\infty$ is l_∞ norm. $(\mathbf{y} \leftarrow \mathcal{D}_\mathbf{y})$ means that $\mathbf{y}$ is chosen according to some distribution $\mathcal{D}_\mathbf{y}$ (uniform or Gaussian) at random. If $\mathbf{y} \leftarrow \mathcal{D}_\mathbf{y}$, it means that $\|\mathbf{y}\|_\infty \leq \gamma$ using uniform sampling. Using Gaussian sampling, we denote $\mathbf{y} \leftarrow \mathcal{D}_\sigma$, where σ is the standard deviation.

4.2. Definitions of Linearly Homomorphic Signature

Definition 2. Given a fixed ring $\mathcal{R}$, a linearly homomorphic signature over it contains a tuple of probabilistic polynomial-time algorithms (Setup, Sign, Verify, Combine) and the detailed descriptions can be seen as follows:

- (1) **Setup** (n, params) . It is a probabilistic algorithm that outputs (pk, sk) by inputting a security parameter n and other public parameters (params) .
- (2) **Sign** (μ, sk, τ) . It is a probabilistic algorithm that outputs a valid signature σ by inputting secret key (sk) , a basis vector μ of message set $(\mathcal{M} (\mu \in \mathcal{M} \subset \mathbb{F}_2^n))$, and a tag (or an identifier id) $(\tau \in \{0, 1\}^*)$ of message $\mathcal{M}$.
- (3) **Verify** $(\mu, \text{pk}, \sigma, \tau)$. It is a deterministic algorithm that outputs a bit b by inputting the tuple $(\mu, \text{pk}, \sigma, \tau)$. If σ is a valid signature of μ , the algorithm outputs $b = 1$ (accept); otherwise, $b = 0$ (reject).
- (4) **Combine** $(\text{pk}, \tau, \{a_i, \mu_i, \sigma_i\}_{i=1}^L)$. This algorithm outputs a valid combined signature (σ') of $(\mu' = \sum_{i=1}^L a_i \mu_i \bmod 2)$, where $(a_i \in \{0, 1\}, i \leq L)$. The parameter L is the maximum circuit depth.

In general, the security properties of a linearly homomorphic signature scheme contain correctness,

unforgeability, and privacy. We will give the specific contents for them as follows:

- (1) **Correctness:** the outputs from above algorithms Sign and Combine can be accepted by the Verify algorithm.
- (2) **Unforgeability:** we will show a game between challenger $\mathcal{C}$ and a polynomial-time adversary $\mathcal{A}$.
 - (1) **Setup:** the challenger $\mathcal{C}$ runs algorithm $(\text{Setup}(n, \text{params}))$ to get (pk, sk) and gives (pk) to the adversary $\mathcal{A}$.
 - (2) **Sign queries:** the adversary $\mathcal{A}$ makes adaptive signature queries on k -dimensional subspaces $\mathcal{U}_i$ of message space $\mathcal{M}$, and he chooses a basis vectors $(\mu_{i1}, \dots, \mu_{ik})$ for $\mathcal{U}_i$. For each subspaces $\mathcal{U}_i$, the challenger $\mathcal{C}$ chooses τ_i from $\{0, 1\}^n$ at random and gives τ_i and j signatures $(\sigma_{ij} \leftarrow \text{Sign}(\text{sk}, \tau_i, \mu_{ij}))$ to the adversary $\mathcal{A}$, where $j = 1, 2, \dots, k$.
 - (3) **Output:** the adversary $\mathcal{A}$ outputs a tag τ^* , a nonzero message $\mathcal{U}^*$, and a signature σ^* .

The adversary wins the game when his outputs satisfy the algorithm

$$\text{Verify}(\mathcal{U}^*, \text{pk}, \sigma^*, \tau^*) = 1, \quad (2)$$

and this algorithm satisfies the following one of two conditions:

- (1) Type 1. $(\tau^* \neq \tau_i)$ for all i .
- (2) Type 2. $(\tau^* = \tau_i)$ but $(\mathcal{U}^* \notin \mathcal{U}_i)$.

Definition 3. A linearly homomorphic signature scheme (Setup, Sign, Verify, Combine) is unforgeability if the probability advantage of adversary winning above game is negligible with security parameter n . That is,

$$\left| \Pr[\text{Verify}(\mathcal{U}^*, \text{pk}, \sigma^*, \tau^*) = 1 \mid \tau^* \neq \tau_i] \right| \leq \varepsilon(n),$$

$$\text{or } \left| \Pr[\text{Verify}(\mathcal{U}^*, \text{pk}, \sigma^*, \tau^*) = 1 \mid \tau^* = \tau_i, \mathcal{U}^* \notin \mathcal{U}_i] \right| \leq \varepsilon(n). \quad (3)$$

- (3) **Privacy:** a game between challenger $\mathcal{C}$ and a polynomial-time adversary $\mathcal{A}$ is shown as follows:

- (1) **Setup:** the challenger $\mathcal{C}$ runs algorithm Setup $(n, (\text{params}))$ to get (pk, sk) and gives (pk) to the adversary $\mathcal{A}$.
- (2) **Challenge:** the adversary $\mathcal{A}$ chooses two linear message subspaces $\mathcal{U}_0$ and $\mathcal{U}_1$ represented as vectors $(\mu_1^{(b)}, \dots, \mu_k^{(b)})$ for $(b = 0, 1)$. In addition, he selects functions $(f_1, \dots, f_s)$, which satisfy $(f_i(\mu_1^{(0)}, \dots, \mu_k^{(0)}) = f_i(\mu_1^{(1)}, \dots, \mu_k^{(1)}))$, where $(i = 1, 2, \dots, s)$.
- (3) **Response:** the challenger $\mathcal{C}$ chooses a random bit $(b \in \{0, 1\})$, a tag $(\tau \in \{0, 1\}^n)$, and signs the vector space $\mathcal{U}_b$. Then, the challenger uses the combine algorithm to generate signatures σ_i for $(f_i(\mu_1^{(b)}, \dots, \mu_k^{(b)}))$ and sends σ_i to $\mathcal{A}$.

TABLE 1: Comparison.

	Public key size	Signature size	Dimension	Resist side-channel attacks
[15]	$(m'n + m'n \log q)$	$(2m' + 2m' \log q + n)$	$(m' \geq 5n \log q)$	×
[16]	$(m'n \log q)$	$(m' \log q + n)$	$(m' \geq 5n \log q)$	×
Ours	$(mn \log q)$	$(2m \log q + n)$	$(m \geq n \log q)$	✓

- (4) **Outputs:** $\mathcal{A}$ outputs a guess bit b' . If $b' = b$ holds, the adversary $\mathcal{A}$ succeeds in the game.

Definition 4. A linearly homomorphic signature scheme (Setup, Sign, Verify, Combine) is privacy if the probability advantage of adversary winning above game is negligible with security parameter n . That is, $|\Pr[b' = b] - (1/2)| \leq \varepsilon(n)$.

5. Our Lattice-Based Linearly Homomorphic Signature

Setup. ($\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$) and ($\mathbf{T} = \mathbf{AS} \bmod q$), where the public key is $(\mathbf{T}, \mathbf{A})$ and secret key is $\mathbf{S}$. We denote the hash function as $(H: \{0, 1\}^* \rightarrow \mathbb{Z}_q^m)$ and another hash function $(h_\alpha(\boldsymbol{\mu}) = \langle \alpha, \boldsymbol{\mu} \rangle \bmod q)$. Obviously, this function satisfies a property $(\sum_{i=1}^m a_i h_\alpha(\boldsymbol{\mu}_i) = h_\alpha(\sum_{i=1}^m a_i \boldsymbol{\mu}_i), a_i \in \mathbb{Z})$. Especially, we assume $(a_i \in \{0, 1\})$ in our scheme as below.

Sign: we suppose the message satisfies $(\boldsymbol{\mu} \in \mathbb{Z}_2^m)$ and choose a basis of it, that is, $\{\boldsymbol{\mu}_1, \dots, \boldsymbol{\mu}_m\}$ (for the sake of design, we have assumed that it is a full-rank space). In addition, the used linear function is $(f(\boldsymbol{\mu}) = \sum_{j=1}^m a_j \boldsymbol{\mu}_j)$. Then, signer does the following steps:

- (1) He chooses $(\mathbf{y}_i \leftarrow_R D_{\mathbf{y}}^m)$ ($\gamma < q$) and computes m vectors $(\alpha_i = H(\mathbf{A}\mathbf{y}_i \bmod q, \tau))$, where $(\tau \in \{0, 1\}^m)$ is a tag of message basis vector $(\boldsymbol{\mu}_j (1 \leq j \leq m))$.
- (2) He computes $(h_{\alpha_j}(\boldsymbol{\mu}_j) = h_{ij} = \langle \alpha_j, \boldsymbol{\mu}_j \rangle \bmod q)$. Then, fixing the parameter j for any message $\boldsymbol{\mu}_j$, he denotes a vector $(\mathbf{h}_j = (h_{1j}, \dots, h_{mj}, \dots, h_{mj})^T)$.
- (3) He computes $(\mathbf{z}_j = \mathbf{S}\mathbf{h}_j + \mathbf{y}_j)$. If $(\|\mathbf{z}_j\|_\infty \leq \gamma - \beta)$, output the signature $(\mathbf{z}_j, \mathbf{h}_j, \tau)$, or else, go to the first step. Here, $(\mathbf{y}_j = \mathbf{y}_i)$ holds.

Verify: the verifier verifies the conditions as follows:

- (1) He computes $(\alpha_j = H(\mathbf{A}\mathbf{z}_j - \mathbf{T}\mathbf{h}_j \bmod q, \tau))$.
- (2) He computes whether the equation $(h_{jj} = \langle \alpha_j, \boldsymbol{\mu}_j \rangle \bmod q)$ holds or not.
- (3) $\|\mathbf{z}_j\|_\infty \leq \gamma - \beta$.

Combine: given public key $\mathbf{A}$ and an array $(a_j, \boldsymbol{\mu}_j, \mathbf{z}_j, \mathbf{h}_j)$ for $(j = 1, \dots, m)$. This algorithm

outputs signature $(\sum_{j=1}^m a_j \mathbf{z}_j, \sum_{j=1}^m a_j \mathbf{h}_j)$ of message $(\sum_{j=1}^m a_j \boldsymbol{\mu}_j)$.

6. Proof of Security

6.1. Correctness. Since correctness refers to two verifications from the outputs of **Sign** and **Combine** algorithms, we prove it one by one:

- (1) The signature from **Sign** algorithm is valid. For each j , when the verifier receives the signature $(\mathbf{z}_j, \mathbf{h}_j, \tau)$, he computes

$$\begin{aligned}
 \alpha_j &= H(\mathbf{A}\mathbf{y}_j \bmod q, \tau) \\
 &= H(\mathbf{A}(\mathbf{z}_j - \mathbf{S}\mathbf{h}_j) \bmod q, \tau) \\
 &= H(\mathbf{A}\mathbf{z}_j - \mathbf{A}\mathbf{S}\mathbf{h}_j \bmod q, \tau) \\
 &= H(\mathbf{A}\mathbf{z}_j - \mathbf{T}\mathbf{h}_j \bmod q, \tau).
 \end{aligned} \tag{4}$$

Then, he computes whether the equation $(\langle \alpha_j, \boldsymbol{\mu}_j \rangle \bmod q = h_{ij})$ holds or not.

- (2) The signature from **Combine** algorithm is valid. We let matrix $\mathbf{H}$ be a composition of vectors $(\alpha_j (1 \leq j \leq m))$; that is, $(\mathbf{H} = (\alpha_1, \alpha_2, \dots, \alpha_m)^T)$. Hence, we have $(\mathbf{h}_j = \mathbf{H}\boldsymbol{\mu}_j \bmod q)$. Since condition $(\alpha_j = H(\mathbf{A}\mathbf{y}_j, \tau))$ holds for each j , we only need to verify the linear property of $(\sum_{j'=1}^m a_{j'} h_{jj'})$ and linear bound of $(\sum_{j=1}^m a_j \mathbf{z}_j)$. At first, we consider the following equation:

$$\begin{aligned}
 \sum_{j'=1}^m a_{j'} \mathbf{h}_{j'} &= \sum_{j'=1}^m a_{j'} \mathbf{H}\boldsymbol{\mu}_{j'} \bmod q \\
 &= \mathbf{H} \left(\sum_{j'=1}^m a_{j'} \boldsymbol{\mu}_{j'} \right) \bmod q.
 \end{aligned} \tag{5}$$

Thus, $(\sum_{j'=1}^m a_{j'} h_{jj'} = \langle \alpha_j, \sum_{j'=1}^m a_{j'} \boldsymbol{\mu}_{j'} \rangle \bmod q)$ holds because of $(h_{jj'} = \langle \alpha_j, \boldsymbol{\mu}_{j'} \rangle \bmod q)$ when $(\alpha_j = H(\mathbf{A}\mathbf{z}_j - \mathbf{T}\mathbf{h}_j \bmod q, \tau))$ holds. Next, we can see that the bound of our signature size is linear obviously. That is,

$$\left\| \sum_{j'=1}^m a_{j'} \mathbf{z}_{j'} \right\|_\infty \leq m \|\mathbf{z}_{j'}\|_\infty. \tag{6}$$

Hence, as long as this in equation holds, the signature is accepted.

6.2. Unforgeability

Theorem 1. *Our scheme is unforgeability if the lattice problem SIS is hard.*

Proof. We suppose that the defined unforgeability game is correctly performed between a challenger $\mathcal{C}$ and a polynomial-time adversary $\mathcal{A}$. In addition, $q_{H,h}$ and q_{sig} are the times of random oracle and signature oracle. Specifically, given the public key $(\mathbf{A}, \mathbf{T})$, $\mathcal{A}$ adaptively chooses some m -dimensional subspaces $\mathcal{U}_{i(n)}$ and chooses basis $(\mu_{i1}, \dots, \mu_{im})$ for $\mathcal{U}_i$. To return m signatures to $\mathcal{A}$, the challenger makes query to above oracles and outputs $(\sigma_{i1}, \dots, \sigma_{im})$ for the chosen basis.

When above game is finished, the adversary $\mathcal{A}$ outputs a tag τ^* , a nonzero message $\mathcal{U}^*$, and a signature σ^* . Next, we consider two types forgeries, respectively. $\square$

Type 1. If $(\tau^* \neq \tau_i)$ for all i and $\text{Verify}((\mathcal{U}^*, \text{pk}, \sigma^*, \tau^*) = 1)$ holds, the adversary has ability to solve the SIS problem. Suppose $\mathcal{A}$ has gotten $((\mathbf{z}_{ij}, \mathbf{h}_{ij}), j = 1, \dots, m)$, then he chooses (α_{ij}^*) randomly and computes $(\mathbf{h}_{ij}^*)$. Hence, he can combine equations as follows:

$$\begin{aligned} \mathbf{z}_{ij} &= \mathbf{S}\mathbf{h}_{ij} + \mathbf{y}_{ij}, \\ \mathbf{z}_{ij}^* &= \mathbf{S}\mathbf{h}_{ij}^* + \mathbf{y}_{ij}. \end{aligned} \quad (7)$$

Thus, he gets $(\mathbf{z}_{ij} - \mathbf{z}_{ij}^* = \mathbf{S}(\mathbf{h}_{ij} - \mathbf{h}_{ij}^*))$ and computes

$$\begin{aligned} \mathbf{A}(\mathbf{z}_{ij} - \mathbf{z}_{ij}^*) &= \mathbf{A}\mathbf{S}(\mathbf{h}_{ij} - \mathbf{h}_{ij}^*) \\ \mathbf{A}\Delta\mathbf{z} &= \mathbf{T}\Delta\mathbf{h}. \end{aligned} \quad (8)$$

Notice that if $(\Delta\mathbf{h} \neq 0)$, the adversary forgeries a signature if and only is he can solve SIS instance. Furthermore, since $|\Pr[\text{Verify}(\mathcal{U}^*, \text{pk}, \sigma^*, \tau^*) = 1 \mid \tau^* \neq \tau_i]| = (|\Pr[\Delta\mathbf{h} = 0 \mid \alpha_{ij}^* \leftarrow_R \mathbb{Z}_q^m]|) = (1/q_{H,h}) + (1/q_{\text{sig}}) \leq \varepsilon(n)$, we can see that the probability of success for this type of attack is negligible.

Type 2. If conditions $(\tau^* = \tau_i)$ and $(\mathcal{U}^* \notin \mathcal{U}_i)$ hold, the adversary also can solve the SIS problem. In this case, for the same hash value (α_{ij}) , the adversary chooses message space $\mathcal{U}^*$ and one of its basis $(\mu_j^* \notin \mathcal{U}_i)$. Then, he computes $(\langle \alpha_{ij}, \mu_j^* \rangle \bmod q)$ and $\mathbf{h}_j^*$. Since (α_{ij}) is a fixed value, the adversary can compute:

$$\begin{aligned} \alpha_j &= H(\mathbf{A}\mathbf{y}_{ij}, \tau) \\ &= H(\mathbf{A}\mathbf{z}_{ij} - \mathbf{T}\mathbf{h}_{ij}, \tau) \\ &= H(\mathbf{A}\mathbf{z}_j^* - \mathbf{T}\mathbf{h}_j^*, \tau). \end{aligned} \quad (9)$$

Then, he obtains equation $(\mathbf{A}(\mathbf{z}_{ij} - \mathbf{z}_j^*) = \mathbf{T}(\mathbf{h}_{ij} - \mathbf{h}_j^*))$, which is marked as $(\mathbf{A}\Delta\mathbf{z} = \mathbf{T}\Delta\mathbf{h})$.

Since $|\Pr[\text{Verify}(\mathcal{U}^*, \text{pk}, \sigma^*, \tau^*) = 1 \mid \tau^* = \tau_i, \mathcal{U}^* \notin \mathcal{U}_i]| = |\Pr[\Delta\mathbf{z} = 0 \mid \mathbf{z}_j^* \leftarrow_R \mathbb{Z}_q^m, \mu_j^* \notin \mathcal{U}_i]| = (1/q_{\text{sig}}) \leq \varepsilon(n)$, the adversary cannot forge a valid signature; otherwise, he is able

to search a SIS problem solution. In addition, the reason why he does not use hash oracle is that the condition $(\tau^* = \tau_i)$ determines this oracle has the same input.

6.3. Privacy

Theorem 2. *Our scheme is privacy.*

Proof. According to the definition of privacy game, challenger $\mathcal{C}$ and adversary $\mathcal{A}$ firstly finish the setup step. Then, $\mathcal{A}$ chooses two basis vectors $(\mu_1^{(b)}, \dots, \mu_m^{(b)})$ from message spaces $\mathcal{U}_b$, where $(b = 0, 1)$. At the same time, he selects linear functions $(f_i (i = 1, 2, \dots, s(n)))$ satisfying $(f_i(\mu_1^{(0)}, \dots, \mu_m^{(0)}) = f_i(\mu_1^{(1)}, \dots, \mu_m^{(1)}))$. Specifically, for a fixed i , the condition $(\sum_{j=1}^m a_j^i \mu_j^{(0)} = \sum_{j=1}^m a_j^i \mu_j^{(1)})$ holds.

When challenger $\mathcal{C}$ obtains $(\mu_1^{(b)}, \dots, \mu_m^{(b)})$, he chooses b and $\tau \in \{0, 1\}^m$ randomly. Then, he signs basis vectors under f_i and outputs the signature σ_i computed using the Com-bine algorithm. And he sends σ_i to $\mathcal{A}$.

Finally, the adversary outputs a guess bit b' . Next, we will show that it is negligible for him to succeed in this game. That is, $|\Pr[b' = b] - 1/2| \leq \varepsilon(n)$.

In fact, we let two distinguished signatures are $(\sigma_i^{(0)} = (\sum_{j=1}^m a_j^i \mu_j^{(0)}, \sum_{j=1}^m a_j^i \mathbf{h}_j^{(1)}))$ of message $(\sum_{j=1}^m a_j^i \mu_j^{(0)})$, $(\sigma_i^{(1)} = (\sum_{j=1}^m a_j^i \mu_j^{(1)}, \sum_{j=1}^m a_j^i \mathbf{h}_j^{(1)}))$ of message $(\sum_{j=1}^m a_j^i \mu_j^{(1)})$, and condition is $(\text{Con} = \sum_{j=1}^m a_j^i \mu_j^{(0)} = \sum_{j=1}^m a_j^i \mu_j^{(1)})$. Then, we have

$$\begin{aligned} \mathbf{H}\left(\sum_{j=1}^m a_j^i \mu_j^{(0)}\right) &= \mathbf{H}\left(\sum_{j=1}^m a_j^i \mu_j^{(1)}\right) \bmod q \left(\sum_{j=1}^m a_j^i \mathbf{h}_j^{(0)}\right) \\ &= \left(\sum_{j=1}^m a_j^i \mathbf{h}_j^{(1)}\right) \bmod q, \end{aligned} \quad (10)$$

where $\mathbf{H} = (\alpha_1, \alpha_2, \dots, \alpha_m)^T$. Hence, we get

$$\left| \Pr[\sigma_i^{(0)} \mid \text{Con}] - \Pr[\sigma_i^{(1)} \mid \text{Con}] \right| \leq \varepsilon(n). \quad (11)$$

Because $(\sigma_i^{(0)})$ and $(\sigma_i^{(1)})$ are indistinguishable under (Con) for the adversary, we show $(|\Pr[b' = b] - (1/2)| \leq \varepsilon(n))$ holds. $\square$

7. Efficiency Comparisons

In schemes of hash-and-sign type [15, 16], the signer uses trapdoor basis to compute preimage sampling function to create signature σ . However, the lattice dimension m must satisfy $(m \geq 5n \log q)$ to get a trapdoor basis (see [17, 18]) and a larger m will result in a larger size of public key and signature.

Our design using the Fiat-Shamir signature framework without trapdoor has smaller public key and signature sizes mainly because $(m \geq n \log q)$ is enough. Then, compared to public key size $(m'n \log q \approx 5n^2 (\log q)^2)$ in [16], our result $(mn \log q \approx n^2 (\log q)^2)$ is smaller than theirs. In addition, signature size of our scheme $(2m \log q + n \approx 2n (\log q)^2 + n)$ is also smaller than $(m' \log q + n \approx 5n (\log q)^2 + n)$ in [16].

Furthermore, preimage sampling function utilizes Gaussian distribution which cannot resist partial side-channel attacks, and we use uniform distribution of aborting technology to resist such attacks effectively. The detailed comparisons can be seen in Table 1.

8. Conclusion and Further Work

8.1. Conclusion. In this paper, we provide a new lattice-based linearly homomorphic signature scheme over $\mathbb{F}_2$ based on the SIS problem. Since we use Fiat-Shamir signature framework instead of hash-and-sign signature framework to design this signature scheme, we do not need to construct a trapdoor basis, and then the whole design is simpler than the existing schemes. At the same time, without the trapdoor basis, our scheme has the smallest public key size ($n^2(\log q)^2$) and signature size ($2n(\log q)^2 + n$) in the existing schemes because of parameter m satisfying ($m \approx n \log q$) rather than ($m \approx 5n \log q$). In addition, under the Fiat-Shamir framework, the use of filtering technology with uniform sampling can resist side-channel attacks.

8.2. Further Work. Decreasing interaction and storage costs is the main work of our future research. In fact, new compression skill and decreasing parameters m and n must be improved efficiency. Since our scheme can be designed on R-SIS directly, we no longer give a special scheme. That is, if each element chosen forms the ring ($\mathcal{R} = (\mathbb{Z}[x]/f(x))$) or ($\mathcal{R}_q = (\mathbb{Z}_q[x]/f(x))$), where ($f(x) = x^n + 1$), n is power of 2, and q is prime, then the parameter m only needs to satisfy $m \approx \log q$, rather than $m \approx n \log q$ for SIS. Hence, it can improve the efficiency.

Specifically, we focus on [9], where it also uses filtering technology (uniform distribution), and special compression methods are used. Meanwhile, module lattice form brings an advantage to parameters m and n , which can be the 1/4 of existing set. In addition, this form can be transformed into lattice hard problem over ring (R-SIS) and general problem (SIS) by setting relative parameters.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (grant nos. 61602287, 11531008, 11771252, and 61632020), State Key Program of National Natural Science of China, Natural Science Foundation of Shandong Province (grant no. ZR2017MF021), Major Innovation Project of Science and Technology, Shandong (grant no. 2018CXGC0702), Primary Research & Development Plan of Shandong Province (grant no.

2018GGX101037), National Innovation Demonstration Zone Development and Construction Fund Project of Shandong Peninsula (grant no. S190101010001), Innovative Research Team in University by Ministry of Education (grant no. IRT16R43), and Taishan Scholars Project.

References

- [1] D. Boneh, D. M. Freeman, J. Katz, and B. Waters, "Signing a linear subspace: signature schemes for network coding," in *Proceedings of the International Conference on Practice and Theory in Public Key Cryptography PKC*, S. Jarecki and G. Tsudik, Eds., pp. 68–87, Irvine, CA, USA, March 2009.
- [2] D. X. Charles, K. Jain, and K. E. Lauter, "Signatures for network coding," *IACR Cryptology*, vol. 25, 2006, <http://eprint.iacr.org/2006/025>.
- [3] R. Gennaro, J. Katz, H. Krawczyk, and T. Rabin, "Secure network coding over the integers," in *Proceedings of the International Conference on Practice and Theory in Public Key Cryptography PKC*, P. Q. Nguyen and D. Pointcheval, Eds., Paris, France, May 2010.
- [4] F. Zhao, T. Kalker, M. Médard, and K. J. Han, "Signatures for content distribution with network coding," in *Proceedings of the IEEE International Symposium on Information Theory ISIT*, pp. 556–560, IEEE, Cambridge, MA, USA, July 2007.
- [5] M. Ajtai, "Generating hard instances of lattice problems (extended abstract)," in *Proceedings of the Twenty-Eighth Annual ACM Symposium on the Theory of Computing*, G. L. Miller, Ed., pp. 99–108, ACM, Philadelphia, PA, USA, May 1996.
- [6] C. Gentry, C. Peikert, and V. Vaikuntanathan, "Trapdoors for hard lattices and new cryptographic constructions," in *Proceedings of the 40th Annual ACM Symposium on Theory of Computing*, C. Dwork, Ed., pp. 197–206, ACM, Victoria, Canada, May 2008.
- [7] P.-A. Fouque, J. Hoffstein, and P. Kirchner, "Falcon: fast-fourier lattice-based compact signatures over nuru," in *Post-quantum Cryptography, NIST, Round 2 Summissions*, Springer, Berlin, Germany, 2018.
- [8] L. Ducas, A. Durmus, T. Lepoint, and V. Lyubashevsky, "Lattice signatures and bimodal Gaussians," in *Proceedings of the 33rd Annual Cryptology Conference (CRYPTO 2013)*, R. Canetti and J. A. Garay, Eds., Santa Barbara, CA, USA, August 2013.
- [9] L. Ducas, T. Lepoint, V. Lyubashevsky, P. Schwabe, G. Seiler, and D. Stehlé, "Crystals-dilithium: digital signatures from module lattices," *IACR Cryptology*, vol. 633, 2017.
- [10] V. Lyubashevsky, "Fiat-Shamir with aborts: applications to lattice and factoring-based signatures," in *Proceedings of the 15th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT 2009)*, M. Matsui, Ed., Tokyo, Japan, December 2009.
- [11] V. Lyubashevsky, "Lattice signatures without trapdoors," in *Proceedings of the 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT 2012)*, D. Pointcheval and T. Johansson, Eds., Cambridge, UK, April 2012.
- [12] L. G. Bruinderink, A. Hülsing, T. Lange, and Y. Yarom, "Flush, gauss, and reload—a cache attack on the BLISS lattice-based signature scheme," in *Proceedings of the 18th International Conference Cryptographic Hardware and Embedded Systems (CHES 2016)*, B. Gierlichs and A. Y. Poschmann, Eds., Santa Barbara, CA, USA, August 2016.

- [13] P. Pessl, "Analyzing the shuffling side-channel countermeasure for lattice-based signatures," in *Proceedings of the 17th International Conference on Cryptology in India Progress in Cryptology INDOCRYPT 2016*, O. Dunkelman and S. K. Sanadhya, Eds., Kolkata, India, December 2016.
- [14] D. Micciancio and M. Walter, "Gaussian sampling over the integers: efficient, generic, constant-time," in *Proceedings of the 37th Annual International Cryptology Conference Advances in Cryptology (CRYPTO 2017)*, J. Katz and H. Shacham, Eds., Santa Barbara, CA, USA, August 2017.
- [15] D. Boneh and D. M. Freeman, "Linearly homomorphic signatures over binary fields and new tools for lattice-based signatures," in *Proceedings of the International Conference on Practice and Theory in Public Key Cryptography PKC*, D. Catalano, N. Fazio, R. Gennaro, and A. Nicolosi, Eds., pp. 1–16, Taormina, Italy, March 2011.
- [16] F. H. Wang, H. U. Yupu, and B. C. Wang, "Lattice-based linearly homomorphic signature scheme over binary field," *Science China (Information Sciences)*, vol. 11, pp. 238–246, 2013.
- [17] M. Ajtai, "Generating hard instances of the short basis problem," in *Proceedings of the 26th International Colloquium Automata, Languages and Programming (ICALP'99)*, J. Wiedermann, P. van Emde Boas, and M. Nielsen, Eds., Prague, Czech Republic, July 1999.
- [18] D. Micciancio and S. Goldwasser, "Complexity of lattice problems-a cryptographic perspective," *The Kluwer International Series in Engineering and Computer Science*, Springer, Berlin, Germany, 2002.
- [19] V. Lyubashevsky, "Lattice-based identification schemes secure under active attacks," in *Proceedings of the 11th International Workshop on Practice and Theory in Public-Key Cryptography (PKC)*, pp. 162–179, Springer, Barcelona, Spain, March 2008.
- [20] M. Rückert, "Lattice-based blind signatures," in *Proceedings of the 16th International Conference on the Theory and Application of Cryptology and Information Security (ASIACRYPT)*, pp. 413–430, Springer, Singapore, December 2010.