A New Fully Homomorphic Signatures from Standard Lattices

Yukun Wang and Mingqiang Wang(✉)

School of Mathematics, Shandong University, Jinan, China
wangyukun@mail.sdu.edu.cn, wangmingqiang@sdu.edu.cn

Abstract. Recently, Gorbunov, Vaikuntanathan and Wichs [6] propose a new powerful (fully) homomorphic trapdoor function (HTDF) based on *small integer solution* (SIS) problem in standard lattices, and construct the first fully homomorphic signature (FHS) schemes. Later Wang et al. [10] extend the notion of HTDF to identity-based setting with strongly security and construct the first identity based fully homomorphic signature (IBFHS) schemes.

In this paper, we provide a new IBHTDF which satisfies *claw-free* and *collision-resistant*. Moreover, we find a homomorphic algorithm for our new IBHTDF where the noise level of multiplication gate is the same as that of addition gate. So, the noise level of IBHTDF for evaluating a circuit of depth d is reduced from $O(4^d m\beta)$ to $O(2^d\beta)$. Finally, we construct a new leveled strongly-unforgeable identity-based fully homomorphic signature (IBFHS) schemes based on our IBHTDF.

Keywords: Identity-based homomorphic trapdoor function · Small integer solution · Strong unforgeability

1 Introduction

In recent years, with the rapid development of cloud computing, a large number of researchers pay more attention to the cryptographic scheme with homomorphic property. The property allows a client to upload his/her encrypted/signed data to a remote server securely. Then, the client could use the computation ability of the server to help him process data but doesn't worry about data leakage. The study of fully homomorphic encryption (FHE) [5], demonstrates how to perform homomorphic computation over encrypted data without the knowledge of secret key, has a far-reaching influence on the latter research. The recent works [1,4,5] of (leveled) fully homomorphic signatures show that how to do homomorphic computation on signed data.

In this work, we focus on the latter question: the public authenticity of the result of homomorphic computation over signed data. In a homomorphic signature scheme, a client signs a message $\mathbf{x} = (x_1, \ldots, x_N)$ using his secret key. After

Supported by organization x.

D. Yu et al. (Eds.): WASA 2020, LNCS 12384, pp. 494–506, 2020.
https://doi.org/10.1007/978-3-030-59016-1_41

that, the client upload the signed data $\sigma = (\sigma_1, \ldots, \sigma_N)$ to a remote server. At any later point, the server obtains an admissible circuit g that $y = g(\mathbf{x})$ and do some homomorphic computation over the signed data σ. In particular, the server produce a short signature σ_g on y which is a correct output of the operation g over the data $\mathbf{x}$. Anyone can verify the tuple (g, y, σ_g) using the client public verification key and accept this fact without the knowledge of the underlying data $\mathbf{x}$.

Leveled FHS. Gorbunov, Vaikuntanathan and Wichs [6] proposed the first leveled FHS schemes based on SIS problem in standard lattices. They put forward a new primitive: HTDF. They required that HTDF functions have *claw-freeness* property, which is necessary for the security of their FHS schemes. Their FHS schemes are existentially unforgeable in the static chosen-message-attack (EU-sCMA) model. Additionally, they showed that one can transform an EU-sCMA secure FHS to an existentially unforgeable under adaptive chosen-message-attack(EU-aCMA) secure FHS via homomorphic chameleon hash function. Recently, Boyen, Fan and Shi also brought up a EU-aCMA secure FHS schemes using vanishing trapdoor technique [3]. In the meantime, Xie and Xue [9] showed that leveled FHS schemes can be constructed if indistinguishability obfuscation and injective one way function exist.

Leveled IBFHS. Wang et al. [10] proposed the first leveled strongly-unforgeable IBFHS schemes. They construct an IBHTDF which is not only *claw-free*, but also *collision-resistant*. They use Barrington's theorem to reduce the parameters as done in field of FHE [2]. The maximum noise-level comparing to Gorbunov, Vaikuntanathan and Wichs' FHS roughly reduces from $O(m^d\beta)$ to $O(4^d m\beta)$, which will result in polynomial modulus $q = \text{poly}(\lambda)$ when $d = O(\log \lambda)$, where λ is the security parameter and d is the maximum depth of admissible circuit.

1.1 Results and Techniques

In this paper, we provide a new IBHTDF and construct a leveled IBFHS based on our IBHTDF. Our new IBFHS scheme is existentially unforgeable in the static chosen-message-attack (EU-sCMA).

For integers n, q and $\ell = \lceil \log q \rceil$, let $\mathbf{G} = \mathbf{I}_n \otimes \mathbf{g}^T \in \mathbb{Z}_q^{n \times n\ell}$, where $\mathbf{g}^T = (1, 2, 2^2, \ldots, 2^{\ell-1})$ and $\mathbf{I}_n$ denotes the n-dimensional identity matrix. The HTDF in [6] is constructed by the function $f_{pk,x} = \mathbf{A} \cdot \mathbf{U} + x \cdot \mathbf{G}$, where $\mathbf{A}$ is a matrix with a trapdoor for invert, and $\mathbf{U}$ is a matrix with small norm. Homomorphic operation relies on the invertibility property of the matrix $\mathbf{G}$. Notice that, if the matrix $\mathbf{G}$ in $f_{pk,x}$ is replaced by the matrix $\mathbf{A}$, one still can evaluate the new function homomorphic. Therefore, the function $f_{pk,x} = \mathbf{A} \cdot \mathbf{U} + x \cdot \mathbf{A}$ is a new HTDF. The homomorphic operation algorithm of our new HTDF is as following.

Homomorphic Operations. Let $\mathbf{U}_1, \mathbf{U}_2 \in \mathbb{Z}_q^{m \times m}$ be "short" matrices and

$$\mathbf{V}_1 = f_{pk,x_1}(\mathbf{U}_1) = \mathbf{A}\mathbf{U}_1 + x_1 \cdot \mathbf{A} \ , \ \mathbf{V}_2 = f_{pk,x_2}(\mathbf{U}_2) = \mathbf{A}\mathbf{U}_2 + x_2 \cdot \mathbf{A}.$$

Addition. We can simply set $\mathbf{U}^* := \mathbf{U}_1 + \mathbf{U}_2,\ \mathbf{V}^* := \mathbf{V}_1 + \mathbf{V}_2$ and get

$$f_{pk,x_1+x_2}(\mathbf{U}^*) = \mathbf{A}\mathbf{U}^* + (x_1 + x_2)\mathbf{A} = \mathbf{V}^*.$$

Multiplication. Homomorphic multiplication is slightly more complex. We set

$$\mathbf{U}^* := \mathbf{U}_1 + x_1 \cdot \mathbf{U}_2,\ \mathbf{V}^* := \mathbf{V}_1 - x_1 \cdot \mathbf{A} + x_1 \cdot \mathbf{V}_2.$$

It is easy to verify

$$f_{pk,x_1 \cdot x_2}(\mathbf{U}^*) = \mathbf{A}\mathbf{U}^* + (x_1 \cdot x_2)\mathbf{A} = \mathbf{V}^*.$$

To evaluate a circuit g of depth d for our new HTDF, the maximum noise level of our algorithm is $O(2^d\beta)$. The homomorphic operation algorithm for the original HTDF require the invert operation of $\mathbf{G}$ which makes the the nose level increasing m multiples. A permutation branching program is used in [10] for evaluating a circuit g of depth d for a new HTDF, that reduce the maximum noise level from $O(m^d\beta)$ to $O(4^d m\beta)$. While, the multiplication operation for our new HTDF does not need invert operation of any matrix. The noise level for our new HTDF of multiplication gate is the same as that of addition gate. So, our noise level should be optimal.

Gorbunov's pioneering work shows that any HTDF must satisfy *claw-free* for security. Later Wang extend the notion of HTDF to IBHTDF with stronger security. The stronger security requires that IBHTDF is not only *claw-free* but also *collision-resistant*. We use a special trapdoor generator which can generates a public matrix with trapdoor for any identity and the function f to construct a new IBHTDF. Because of the new function f, we improve the proving method in [10] to make sure that the new IBHTDF could satisfy *claw-free* and *collision-resistant*.

Finally, we construct a new leveled strongly-unforgeable identity-based fully homomorphic signature (IBFHS) schemes based on our IBHTDF. The maximum noise-level comparing to Wang's FHS [10] roughly reduces from $O(4^d m\beta)$ to $O(2^d\beta)$.

1.2 Paper Organization

In Sect. 2, we give some backgrounds on lattices and related tools used in this paper. We propose the new IBHTDF function in Sect. 3 and demonstrate the homomorphic evaluation algorithm in Sect.4. In Sect. 5 we recall the leveled strongly-unforgeable IBFHS. Finally, we conclude in Sect. 6.

2 Preliminaries

We use the hold upper-case letters (e.g.,**A,B**) to represent matrices and bold lower-case letters (e.g.,**a,b**) to represent column vectors. Let $\|\mathbf{A}\|_\infty = \max_{i,j}\{|a_{i,j}|\}$ denote the infinite norm and a_i or $\mathbf{a}[i]$ represent the i-entry of $\mathbf{a}$. Let $[\mathbf{A}\|\mathbf{B}]$ denote the concatenation of two matrices and $(\mathbf{A},\mathbf{B})$=$[\mathbf{A}^T\|\mathbf{B}^T]^T$. We use λ to denote the *security parameter* and $\mathrm{negl}(\lambda)$ to denote a negligible function that grows slower than λ^{-c} for any constant $c > 0$ and any large enough value of λ. For an integer N, we let $[N] \stackrel{def}{=} \{1, \ldots, N\}$.

2.1 Entropy and Statistical Distance

For discrete random variables $X \leftarrow \mathcal{X}$, $Y \leftarrow \mathcal{Y}$, we define the *statistical distance*

$$\triangle(X,Y) \stackrel{\triangle}{=} \frac{1}{2} \sum_{\omega \in \mathcal{X} \cup \mathcal{Y}} |Pr[X=\omega] - Pr[Y=\omega]|.$$

We say that two random variables X, Y are statistically indistinguishable, denoted by $X \stackrel{stat}{\approx} Y$, if $\triangle(X,Y) = \text{negl}(\lambda)$. The *min-entropy* of a random variable X, denoted by $\mathbf{H}_\infty(X)$, is defined as $\mathbf{H}_\infty(X) \stackrel{\triangle}{=} -\log(max_x \Pr[X=x])$. The *average min-entropy* of X conditioned on Y, denoted by $\mathbf{H}_\infty(X|Y)$, is defined as

$$\mathbf{H}_\infty(X|Y) \stackrel{\triangle}{=} -\log(\mathbf{E}_{y \leftarrow \mathcal{Y}}[max_x \mathbf{Pr}[X=x|Y=y]]) = -\log(\mathbf{E}_{y \leftarrow \mathcal{Y}}[2^{-\mathbf{H}_\infty(X|Y=y)}]).$$

The optimal probability of an unbounded adversary guessing X given the correlated value Y is $2^{-\mathbf{H}_\infty(X|Y)}$.

Lemma 1. *Let $X \leftarrow \mathcal{X}, Y \leftarrow \mathcal{Y}$ be arbitrarily random variables where the support of Y lies in $\mathcal{Y}$. Then $\mathbf{H}_\infty(X|Y) \leq \mathbf{H}_\infty(X) - \log(|\mathcal{Y}|)$.*

2.2 Background on Lattices and Hard Problems

Lattices. Lattices-based cryptography usually use so-called q-ary integer lattices, which contain $q\mathbb{Z}^m$ as a sublattice for some modulus q. Let n, m, q be positive integers. For a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ we define following q-ary integer lattice

$$\Lambda^\perp(\mathbf{A}) = \{\mathbf{u} \in \mathbb{Z}^m : \mathbf{Au} = \mathbf{0} \mod q\}.$$

For a vector $\mathbf{v} \in \mathbb{Z}_q^n$, we define the coset:

$$\Lambda_\mathbf{v}^\perp(\mathbf{A}) = \{\mathbf{u} \in \mathbb{Z}^m : \mathbf{Au} = \mathbf{v} \mod q\}.$$

SIS. Let n, m, q, β be integers. The short integer solution ($\text{SIS}_{n,m,q,\beta}$) problem is that given a uniformly random matrix $\mathbf{A} \stackrel{\$}{\leftarrow} \mathbb{Z}_q^{n \times m}$, to find a nonzero vector $\mathbf{u} \in \mathbb{Z}_q^n$ with $\|\mathbf{u}\|_\infty \leq \beta$ such that $\mathbf{Au} = \mathbf{0} \mod q$(i.e. $\mathbf{u} \in \Lambda^\perp(\mathbf{A})$). For $q \geq \beta \cdot (\sqrt{n \log n})$, the $\text{SIS}_{n,m,q,\beta}$ problem in average case is as hard as solving $\text{GapSVP}_{\widetilde{O}(\beta \cdot \sqrt{n})}$ in the worse case in standard lattices [7,11].

Discrete Gaussian Distribution. Let $\mathcal{D}_{\mathbb{Z}^m,r}$ be the truncated discrete Gaussian distribution over $\mathbb{Z}^m$ with parameter r. That means $\|\mathbf{u}\|_\infty \leq r \cdot \sqrt{m}$ with probability 1 if $\mathbf{u} \leftarrow \mathcal{D}_{\mathbb{Z}^m,r}$. If $\|\mathbf{u}\|_\infty$ is larger than $r \cdot \sqrt{m}$, then the output is replaced by $\mathbf{0}$.

Lattices Trapdoor. Here we recall the trapdoor generation algorithm and Gaussian sampling algorithm in [8]. We ignore all details of implementation which are not strictly necessary in this work.

For integers n, q and $\ell = \lceil \log q \rceil$, let $\mathbf{G} = \mathbf{I}_n \otimes \mathbf{g}^T \in \mathbb{Z}_q^{n \times n\ell}$, where $\mathbf{g}^T = (1, 2, 2^2, \ldots, 2^{\ell-1})$ and $\mathbf{I}_n$ denotes the n-dimensional identity matrix.

Lemma 2. *Let* n, q, ℓ, m_0, m_1 *be integers such that* n=*poly*(λ), $q = q(n)$, $\ell = \lceil \log q \rceil$, $m_0 = n(\ell + O(1))$, $m_1 = n\ell$. *For* $\mathbf{A}_0 \xleftarrow{\$} \mathbb{Z}_q^{n \times m_0}$ *and* $\mathbf{H} \in \mathbb{Z}_q^{n \times n}$, *there exists a randomized algorithm* $\mathbf{TrapGen}(\mathbf{A}_0, \mathbf{H})$ *to generate a matrix* $\mathbf{A} = [\mathbf{A}_0 \| \mathbf{HG} - \mathbf{A}_0\mathbf{R}] \in \mathbb{Z}_q^{n \times (m_0+m_1)}$ *with trapdooor* $\mathbf{R}$ *such that* $\mathbf{R} \leftarrow \mathcal{D}_{\mathbb{Z}^{m_0 \times m_1}, r}$ *for large enough* $r(\geq \omega(\sqrt{\log n}))$ *and* $\mathbf{A}$ *is negl*(λ)*-far from* $(\mathbf{V}_0, \mathbf{V}_1) \xleftarrow{\$} \mathbb{Z}_q^{n \times m_0} \times \mathbb{Z}_q^{n \times m_1}$. *Here,* $\mathbf{R}$ *is called* $\mathbf{G}$*-trapdoor of* $\mathbf{A}$ *with tag* $\mathbf{H}$. *Furthermore, for any non-zero* $\mathbf{u} = (\mathbf{u}_0, \mathbf{u}_1) \in \mathbb{Z}_q^{m_0+m_1}$, *the average min-entropy of* $\mathbf{Ru}_1$ *given* $\mathbf{A}_0$ *and* $\mathbf{A}_0\mathbf{R}$ *is at least* $\Omega(n)$.

Lemma 3. *Given parameters in above lemma and an uniformly random vector* $\mathbf{v} \in \mathbb{Z}_q^n$, *for some* s $(\geq O(\sqrt{n \log q})) \in \mathbb{R}$ *and a fixed function* $\omega(\sqrt{\log n})$ *growing asymptotically faster than* $\sqrt{\log n}$, *if the tag* $\mathbf{H}$ *is invertiable, then there exists an efficient algorithm* $\mathbf{SamPre}(\mathbf{A}_0, \mathbf{R}, \mathbf{H}, \mathbf{v}, s)$ *that samples a vector* $\mathbf{u}$ *from* $\mathcal{D}_{\Lambda_v^\perp(\mathbf{A}), s \cdot \omega(\sqrt{\log n})}$ *such that* $\mathbf{A} \cdot \mathbf{u} = \mathbf{v}$. *Note that* $\|\mathbf{u}\|_\infty \leq s\sqrt{m_0 + m_1} \cdot \omega(\sqrt{\log n})$ *with probability 1. Furthermore, for* $\mathbf{u}' \leftarrow \mathcal{D}_{\mathbb{Z}^m, s \cdot \omega(\sqrt{\log n})}$ *and* $\mathbf{v}' = \mathbf{Au}'$, *we have* $(\mathbf{A}, \mathbf{R}, \mathbf{u}, \mathbf{v}) \overset{stat}{\approx} (\mathbf{A}, \mathbf{R}, \mathbf{u}', \mathbf{v}')$.

3 Identity-Based Homomorphic Signature

In this section, we come up with the definition of IBHTDF and construct a new function f. Based on f, we design a new IBHTDF which satisfy *claw free* and *collision-resistance*. Our IBHTDF is selective-identity secure under the SIS assumption.

3.1 Definition Identity-Based Trapdoor Functions

An *identity-based homomorphic trapdoor function* (IBHTDF) consists of six polynomial algorithms(**IBHTDF.Setup**, **IBHTDF.Extract**, f, **Invert**, **IBHTDF.Eval$^{\mathbf{in}}$**, **IBHTDF.Eval$^{\mathbf{out}}$**) with syntax as follows:

- $(mpk, msk) \leftarrow \mathbf{IBHTDF.Setup}(\mathbf{1}^\lambda)$: Master key setup procedure. The security parameter λ defines the identity space $\mathcal{I}$, the index spacce $\mathcal{X}$, the input space $\mathcal{U}$, the output space $\mathcal{V}$ and some efficiently samplable input distribution $\mathcal{D}_\mathcal{U}$ over $\mathcal{U}$. We require that elements of $\mathcal{I}$,$\mathcal{U}$,$\mathcal{V}$, or $\mathcal{X}$ can be efficiently certified and we can efficiently sample elements from $\mathcal{V}$ uniformly at random.
- $(pk_{id}, sk_{id}) \leftarrow \mathbf{IBHTDF.Extract}(mpk, msk, id)$: An identity-key extraction procedure. We require that pk_{id} can be extracted deterministically from mpk and $id \in \mathcal{I}$ without using the knowledge of msk.
- $f_{pk_{id},x}$: $\mathcal{U} \to \mathcal{V}$: A deterministic function indexed by pk_{id} and $x \in \mathcal{X}$.
- $\mathbf{Invert}_{sk_{id},x} : \mathcal{V} \to \mathcal{U}$: A probability inversion algorithm indexed by sk_{id} and $x \in \mathcal{X}$.
- $u_g = \mathbf{IBHTDF.Eval}^{in}(g, (x_1, u_1, v_1), \ldots, (x_\ell, u_\ell, v_\ell))$: A deterministic *input* homomorphic evaluation algorithm. It takes as input some function $g : \mathcal{X}^\ell \to \mathcal{X}$ and values $\{x_i \in \mathcal{X}, u_i \in \mathcal{U}, v_i \in \mathcal{V}\}_{i \in [\ell]}$ and output $u_g \in \mathcal{U}$.

– $v_g = \mathbf{IBHTDF.Eval}^{out}(g, v_1, \ldots, v_\ell)$: A deterministic *output* homomorphic evaluation algorithm. It takes as input some function $g : \mathcal{X}^\ell \to \mathcal{X}$ and values $\{v_i \in \mathcal{V}\}_{i\in[\ell]}$ and output $v_g \in \mathcal{V}$.

Correctness of Homomorphic Computation. Let algorithm $(pk_{id}, sk_{id}) \leftarrow \mathbf{IBHTDF.Extract}(mpk, msk, id)$ extracts the identity-key for id. Let $g : \mathcal{X}^\ell \to \mathcal{X}$ be a function on $x_1, \ldots, x_\ell \in \mathcal{X}$ and $y = g(x_1, \ldots, x_\ell)$. Let $u_1, \ldots, u_\ell \in \mathcal{U}$ and set $v_i = f_{pk_{id},x_i}(u_i)$ for $i \in [\ell]$. Set $u_g = \mathbf{IBHTDF.Eval}^{in}(g, (x_1, u_1, v_1), \ldots, (x_\ell, u_\ell, v_\ell))$, $v_g = \mathbf{IBHTDF.Eval}^{out}(g, v_1, \ldots, v_\ell)$. We require that $u_g \in \mathcal{U}$ and $v_g = f_{pk_{id},y}(u_g)$.

Distributional Equivalence of Inversion. For the security of our construction IBFHS in next section, we require the following statistical indistinguishability:

$$(pk_{id}, sk_{id}, x, u, v) \overset{stat}{\approx} (pk_{id}, sk_{id}, x, u', v')$$

Where $(pk_{id}, sk_{id}) \leftarrow \mathbf{IBHTDF.Extract}$, $x \in \mathcal{X}$, $u \leftarrow \mathcal{D}_{\mathcal{U}}$, $v = f_{pk_{id},x}(u)$, $v' \xleftarrow{\$} \mathcal{V}$, $u' \leftarrow \mathbf{Invert}_{sk_{id},x}(v')$.

IBHTDF Security. We require not only *claw-freenesss* but also *collision-resistance* for **IBHTDF** security to guarantee *strong-unforgeability* for **IBFHS**.

The experiment $\mathbf{Exp}^{\text{sID}}_{\mathcal{A},\mathbf{IBHTDF}}(1^\lambda)$ describe the selective-*identiy* security, where the adversary has to appoint a target identity id^* to attack before seeing the public key. Moreover, the adversary can query identity-key for all identity except id^*. Then he is required to find $u \neq u' \in \mathcal{U}$, x, $x' \in \mathcal{X}$ such that $f_{pk_{id^*},x}(u) = f_{pk_{id^*},x'}(u')$. It's easy to see that if $x = x'$, then (u, u') is a collision, a claw otherwise.

$$\mathbf{Exp}^{\text{sID}}_{\mathcal{A},\mathbf{IBHTDF}}(1^\lambda)$$

– $(id^*, state) \leftarrow \mathcal{A}(1^\lambda)$.
– $(mpk, msk) \leftarrow \mathbf{IBHTDF.Setup}(1^\lambda)$.
– $(u, u', x, x') \leftarrow \mathcal{A}^{\text{IBHTDF.Extract}(mpk,msk)\setminus id^*}(mpk, state)$.
– $\mathcal{A}$ wins if $u \neq u' \in \mathcal{U}, x, x' \in \mathcal{X}$ are such that $f_{pk_{id^*},x}(u) = f_{pk_{id^*},x'}(u')$.

We say that an identity-based homomorphic trapdoor function is *selective-identity* secure if $\Pr[\mathbf{Exp}^{\text{sID}}_{\mathcal{A},\mathbf{IBHTDF}}(1^\lambda)] \leq \text{negl}(\lambda)$.

3.2 Construction: Basic Algorithms and Security

To describe the **IBHTDF** functions, we give some public parameters as follows.

– Let flexible d be the circuit depth such that $d \leq poly(\lambda)$ and λ be a security parameter.
– Choose an integer $n = \text{poly}(\lambda)$ and a sufficiently large prime $q = q(n)$. Let $\ell = \lceil \log q \rceil$, $m_0 = n(\ell + O(1))$, $m_1 = n\ell$ and $m = m_0 + 2m_1$. Set $\beta_0 = O((n \log\ q)^{3/2})$, $\beta_{max} = O(2^d\beta_0)$, $\beta_{\text{SIS}} = O(m_1\beta_0)\beta_{max} < q$.
– $\mathbf{G} = \mathbf{I_n} \otimes \mathbf{g}^T \in \mathbb{Z}_q^{n\times n\ell}$ is the primitive matrix, where $\mathbf{g}^T = (1, 2, 2^2, \ldots, 2^{\ell-1})$.

- We assume that identities are elements in $\mathrm{GF}(q^n)$, and say $\mathbf{H} : \mathrm{GF}(q^n) \to \mathbb{Z}_q^{n\times n}$ is an invertible difference, if $\mathbf{H}(id_1) - \mathbf{H}(id_2)$ is invertible for any two different identities id_1, id_2 and $\mathbf{H}$ is computable in polynomial time in $n\ell$.
- Set $\mathcal{X} = \mathbb{Z}_2,\ \mathcal{I} = \mathbb{Z}_q^n,\ \mathcal{V} = \mathbb{Z}_q^{n\times m}$ and $\mathcal{U} = \{\mathbf{U} \in \mathbb{Z}_q^{m\times m} : \|\mathbf{U}\|_\infty \le \beta_{\max}\}$. Define the distribution $\mathcal{D}_\mathcal{U}$ is a truncated discrete Gaussian distribution over $\mathcal{U}$, so that $\|\mathbf{U}\|_\infty \le \beta_0$ if $\mathbf{U} \leftarrow \mathcal{D}_\mathcal{U}$.

Now we describe the basic algorithms of **IBHTDF** function $\mathcal{F}$.

- **IBHTDF.Setup**(1^λ): On input a security parameter λ, set $d,\ n,\ m_0,\ m_1,\ m,\ q,\ \beta_0,\ \beta_{max},\ \beta_{SIS}$ as specified above. Then do
  1. Choose $\mathbf{A}_0 \xleftarrow{\$} \mathbb{Z}_q^{n\times m_0}$ and run $\mathbf{TrapGen}(\mathbf{A_0}, \mathbf{0})$ to generate a matrix $\mathbf{A} = [\mathbf{A_0}\|\mathbf{A_1}] = [\mathbf{A_0}\|-\mathbf{A_0}\mathbf{R}]$ and a trapdoor R such that $\mathbf{R} \leftarrow \mathcal{D}_{\mathbb{Z}^{m_0\times m_1},\omega(\sqrt{\log\ n})}$ and $\mathbf{A}$ is negl(λ)-far from uniform. Set the master secret key as $msk = \mathbf{R}$. Note that $\mathbf{A}\cdot(\mathbf{R}, \mathbf{I}_{m_1}) = \mathbf{0}$, namely $\mathbf{R}$ is a $\mathbf{G}$-trapdoor of $\mathbf{A}$ with tag $\mathbf{0}$.
  2. Choose $\mathbf{A}_2 \xleftarrow{\$} \mathbb{Z}_q^{n\times m_1}$ and set public key as $mpk = \{\mathbf{A}, \mathbf{A}_2\}$.
- **IBHTDF.Extract**$(mpk, \mathbf{R}, id)$: On input a master public key mpk, a master secret key $\mathbf{R}$ and an identity $id \in \mathcal{I}$, do
  1. Compute $\mathbf{H}_{id}$ for $id \in \mathcal{I}$ and let $\mathbf{A}'_{id} = [\mathbf{A}_0\|\mathbf{H}_{id}\cdot\mathbf{G} + \mathbf{A}_1]$. Then $\mathbf{R}$ is a $\mathbf{G}$-trapdoor of $\mathbf{A}'_{id}$ with tag $\mathbf{H}_{id}$. Set user's public key $pk_{id} = \mathbf{A}_{id} = [\mathbf{A}'_{id}\|\mathbf{A}_2]$.
  2. Run $\mathbf{SamPre}(\mathbf{A}_0, \mathbf{R}, \mathbf{H}(id), \mathbf{G} - \mathbf{A}_2, O(\sqrt{n\log\ q}))$ to output $\mathbf{R}_{id} \in \mathbb{Z}^{(m_0+m_1)\times m_1}$ such that $\mathbf{A}'_{id}\cdot\mathbf{R}_{id} = \mathbf{G} - \mathbf{A}_2$. Then $\mathbf{R}_{id}$ is a $\mathbf{G}$-trapdoor of $\mathbf{A}_{id}$ with tag $\mathbf{I}_n$. Set secret key $sk_{id} = \mathbf{R}_{id}$.
- $f_{pk_{id},x}(\mathbf{U})$: On input $mpk, id \in \mathcal{I}, x \in \mathcal{X}$ and $\mathbf{U} \in \mathcal{U}$, do
  1. Compute $pk_{id} = \mathbf{A}_{id} = [\mathbf{A}_0\|\mathbf{H}_{id}\cdot\mathbf{G} + \mathbf{A}_1\|\mathbf{A}_2]$ as above.
  2. For $id \in \mathcal{I},\ x \in \mathcal{X}$ and $\mathbf{U} \in \mathcal{U}$, define $f_{pk_{id},x}(\mathbf{U}) \triangleq \mathbf{A}_{id}\cdot\mathbf{U} + x\cdot\mathbf{A}_{id}$.
- $\mathbf{Invert}_{sk_{id},x}(\mathbf{V})$: On input identity $id \in \mathcal{I}$, an identity-key $\mathbf{R}_{id}$, an index $x \in \mathcal{X}$ and $\mathbf{V} \in \mathcal{V}$, run $\mathbf{SamPre}(\mathbf{A}'_{id}, \mathbf{R}_{id}, \mathbf{I}_n, \mathbf{V} - x\cdot\mathbf{A}_{id}, O(n\log\ q))$ to output $\mathbf{U}$ (such that $\mathbf{A}_{id}\cdot\mathbf{U} = \mathbf{V} - x\cdot\mathbf{A}_{id}$).

Distributional Equivalence of Inversion. Let $x \in \mathcal{X}$ and $(pk_{id} = \mathbf{A}_{id}, sk_{id} = \mathbf{R}_{id}) \leftarrow \mathbf{IBHTDF.Extract}\ (mpk, \mathbf{R}, id)$. $\mathbf{U} \in \mathcal{U},\ \mathbf{V} = f_{pk_{id},x}(\mathbf{U}) = \mathbf{A}_{id}\cdot\mathbf{U} + x\cdot\mathbf{A}_{id},\ \mathbf{V}' \xleftarrow{\$} \mathcal{V},\ \mathbf{U}' \leftarrow \mathbf{SamPre}(\mathbf{A}'_{id},\ \mathbf{R}_{id},\ \mathbf{I}_n,\ \mathbf{V}' - x\cdot\mathbf{A}_{id},\ O(n\log\ q))$. By Lemma3 and the fact that $(\mathbf{V}' - x\mathbf{A}_{id})$ is uniformly random, we have

$$(\mathbf{A}_{id}, \mathbf{R}_{id}, \mathbf{U}, \mathbf{A}_{id}\cdot\mathbf{U}) \overset{stat}{\approx} (\mathbf{A}_{id}, \mathbf{R}_{id}, \mathbf{U}', \mathbf{V}' - x\mathbf{A}_{id}).$$

Then, we have

$$(\mathbf{A}_{id}, \mathbf{R}_{id}, x, \mathbf{U}, \mathbf{V} = \mathbf{A}_{id}\cdot\mathbf{U} + x\cdot\mathbf{A}_{id}) \overset{stat}{\approx} (\mathbf{A}_{id}, \mathbf{R}_{id}, x, \mathbf{U}', \mathbf{V}').$$

IBHTDF Security. We now show that the **IBHTDF** function $\mathcal{F}$ is selective-identity secure assuming the SIS assumption.

Theorem 1. *The function $\mathcal{F}$ constructed above is a selective-secure* **IBHTDF** *assuming the* $\mathbf{SIS}_{n,m_0,q,\beta_{SIS}}$.

Because of space limitations, we put the proof of the theorem in the full version.

4 Homomorphic Evaluation and Noise Analysis

In this section, we give a new construction of homomorphic evaluation algorithm. Our construction could do better in homomorphic evaluation based on the fact that the noise growth is slower.

4.1 Basic Homomorphic Evaluation

We now define the basic homomorphic addition and multiplication algorithms that will be used in IBHTDFs. These algorithms for IBHTDFs are simple and faster than that in [10]. But the parameters used in this section are same as that in [10] because of the similar structure. Recall that $\mathbf{V}_i = \mathbf{A}\mathbf{U}_i + x_i\mathbf{A}$ $(i = 1, 2)$, where we set $\mathbf{A} = \mathbf{A}_{id}$ for simplicity throughout Sect. 5. Let $\|\mathbf{U}_i\|_\infty \leq \beta_i$ and $x_i \in \{0, 1\}$.

4.2 Construction: Homomorphic Evaluation and Noise Growth

Now we define the algorithms $\mathbf{Eval}^{in}, \mathbf{Eval}^{out}$ with the syntax

$$\mathbf{U}^* := \mathbf{IBHTDF.Eval}^{in}_{pk}(g, (x_1, \mathbf{U}_1), \ldots, (x_\ell, \mathbf{U}_\ell)),$$

$$\mathbf{V}^* := \mathbf{IBHTDF.Eval}^{out}_{pk}(g, \mathbf{V}_1, \ldots, \mathbf{V}_\ell).$$

We consider the function g as basic gates in an arithmetic circuit: *addition, multiplication, addition-with-constant* and *multiplication-by-constant*. These functions are complete and can be composed to evaluate arbitrary arithmetic circuit. Let the matrice $\mathbf{U}_i$ have noise-levels bounded by β_i.

- Let $g(x_1, x_2) = x_1 + x_2$ be an *addition* gate. The algorithms $\mathbf{IBHTDF.Eval}^{in}$, $\mathbf{IBHTDF.Eval}^{out}$ respectively compute

$$\mathbf{U}^* := \mathbf{U}_1 + \mathbf{U}_2,\ \mathbf{V}^* := \mathbf{V}_1 + \mathbf{V}_2.$$

  The matrix $\mathbf{U}^*$ has noise level $\beta^* \leq \beta_1 + \beta_2$. The correctness follows by $(\mathbf{V}_1 + \mathbf{V}_2) = \mathbf{A}(\mathbf{U}_1 + \mathbf{U}_2) + (x_1 + x_2)\mathbf{A}$.
- Let $g(x_1, x_2) = x_1 \cdot x_2$ be a *multiplication* gate. The algorithms $\mathbf{Eval}^{in}, \mathbf{Eval}^{out}$ respectively compute

$$\mathbf{U}^* := \mathbf{U}_1 + x_1 \cdot \mathbf{U}_2,\ \mathbf{V}^* := \mathbf{V}_1 - x_1 \cdot \mathbf{A} + x_1 \cdot \mathbf{V}_2.$$

  The matrix $\mathbf{U}^*$ has noise level $\beta^* \leq \beta_1 + |x_1|\beta_2 = \beta_1 + \beta_2$. The correctness follows by a simple computation assuming $\mathbf{V}_i = \mathbf{A}\mathbf{U}_i + x_i\mathbf{G}$.
- Let $g(x) = x + a$ be *addition-with constant* gate, for the constant $a \in \mathbb{Z}_q$. The algorithms $\mathbf{IBHTDF.Eval}^{in}$, $\mathbf{IBHTDF.Eval}^{out}$ respectively compute

$$\mathbf{U}^* := \mathbf{U},\ \mathbf{V}^* := \mathbf{V} - a \cdot \mathbf{A}.$$

  The matrix $\mathbf{U}^*$ have the same noise level as $\mathbf{U}$.

– Let $g(x) = a \cdot x$ be a *multiplication-by-constant* gate for the constant $a \in \mathbb{Z}_q$. The algorithms $\mathbf{IBHTDF.Eval}^{in}$, $\mathbf{IBHTDF.Eval}^{out}$ respectively compute

$$\mathbf{U}^* := a \cdot \mathbf{U},\ \mathbf{V}^* := a \cdot \mathbf{V}.$$

The matrix $\mathbf{U}^*$ have the same noise level as $a \cdot \beta$.

Bounded-Depth Circuits. In Wang's IBHTDF, a depth-d ($d \leq \text{poly}(\lambda)$) circuit can be transformed to a length $L = 4^d$ permutation branching program. The maximum noise comparing to Gorbunov-Vaikuntanathan-Wich's HTDF reduces roughly from $O(m^d\beta)$ to $O(4^d m\beta)$. Then in our HTDF we do not use permutation branching program and reduces roughly from $O(m^d\beta)$ to $O(2^d\beta)$. In particular, we can set polynomial modulus $q = \text{poly} > O(2^d\beta)$ when $d = O(\log \lambda)$ which will result in better security based on GapSVP with polynomial approximation factors.

5 Strongly-Unforgeable Identity-Based Fully Homomorphic Signatures

In this section, we give a strongly-unforgeable identity-based fully homomorphic signature scheme. The scheme will take advantage of IBHTDF in Sect. 3 and homomorphic evaluation in Sect. 4.

5.1 Definition of IBFHS

A single data-set identity-based homomorphic signature scheme consists of following algorithms (**PrmsGen**, **Setup**, **Extract**, **Sign**, **SignEval**, **Process**, **Verify**) with syntax:

– $prms \leftarrow \mathbf{PrmsGen}(1^\lambda, 1^\ell)$: Take the security parameter λ and the maximum data-size N. Output public parameters $prms$. The message space $\mathcal{X}$ is defined by security parameter λ.
– $(mpk, msk) \leftarrow \mathbf{Setup}(1^\lambda)$: Take the security parameter λ. Output a master key pair (mpk, msk).
– $(pk_{id}, sk_{id}) \leftarrow \mathbf{Extract}(mpk, msk, id)$: An identity-key extraction procedure.
– $(\sigma_1, \ldots, \sigma_N) \leftarrow \mathbf{Sign}_{sk_{id}}(prms, x_1, \ldots, x_N)$: Sign message data $(x_1, \ldots, x_N) \in \mathcal{X}^N$ for ID.
– $\sigma_g = \mathbf{SignEval}_{prms}(g, (x_1, \sigma_1), \ldots, (x_\ell, \sigma_\ell))$: A deterministic homomorphic signature algorithm output a signature σ_g for some function g over$(x_1, \ldots, x_\ell) \in \mathcal{X}^\ell$.
– $v_g = \mathbf{Process}_{prms}(g)$: Deterministically and homomorphically evaluate a *certificate* v_g for the function g from the public parameters $prms$.
– $\mathbf{Verify}_{pk_{id}}(v_g, y, \sigma_g)$: Verify that y is the correct output of g by proving σ_g corresponding to v_g.

Correctness. For $prms \leftarrow \mathbf{PrmsGen}(1^\lambda, 1^\ell)$, $(pk_{id}, sk_{id}) \leftarrow \mathbf{Extract}(mmpk, msk, id)$, $(x_1, \ldots, x_N) \in \mathcal{X}^N$, $(\sigma_1, \ldots, \sigma_N) \leftarrow \mathbf{Sign}_{sk_{id}}(prms, x_1, \ldots, x_N)$ and $g : \mathcal{X}^\ell \to \mathcal{X}$, we require following equation

$$\mathbf{Verify}_{pk_{id}}(v_g, y = g(x_1, \ldots, x_\ell), \sigma_g) = \text{accept}$$

holds, where $v_g = \mathbf{Process}_{prms}(g)$ and $\sigma_g = \mathbf{Process}_{prms}(g, (x_1, \sigma_1), \ldots, (x_\ell, \sigma_\ell))$.

Correctness of Leveled IBFHS. The correctness of leveled IBFHS follows from that of leveled IBHTDF and hence is omitted.

Security Experiment. The experiment $\mathbf{Exp}^{\text{SU}-\text{sID}-\text{sCMA}}_{\mathcal{A},\text{IBFHS}}(1^\lambda)$ defined in following describes the *strongly-unforgeable selective-identity static chosen-message-attack* security game, where the adversary has to fix a target identity id^* to attack before obtaining the master public-key and public parameters. Moreover the adversary can query identity-keys for all identity except id^*. Then the adversary is forced to find (g, y', σ') such that the **Verify** algorithm output accept. If $y = y'$ then σ' is a strongly-forgeable signature, otherwise is a existentially-forgeable signature.

$$\mathbf{Exp}^{\text{SU}-\text{sID}-\text{sCMA}}_{\mathcal{A},\text{IBFHS}}(1^\lambda)$$

- $(id^*, \{x_i\}_{i\in[N]}, state) \leftarrow \mathcal{A}(1^\lambda)$.
- $prms \leftarrow \mathbf{PrmsGen}(1^\lambda, 1^N), (mpk, msk) \leftarrow \mathbf{Setup}(1^\lambda)$.
- $(g, y', \sigma') \leftarrow \mathcal{A}^{\mathbf{Extract}(mpk,msk,\cdot)\setminus\{id^*\},\mathbf{Sign}(id^*,\{x_i\}_{i\in[N]})}(prms, mpk, state)$.
- $\mathcal{A}$ wins if all of the following hold:
  1. g is a admissible circuit on the messages $x_1, \ldots, x_N$;
  2. $\sigma' \neq \sigma_g$, where $\sigma_g = \mathbf{SignEval}_{prms}(g, (x_1, \sigma_1), \ldots, (x_\ell, \sigma_\ell))$;
  3. $\mathbf{Verify}_{pk_{id^*}}(v_g, y', \sigma')$ accept, where $v_g = \mathbf{Process}_{prms}(g)$.

We say that an IBFHS is *strongly-unforgeable selective-identity static chosen-message-attack* (SU-sID-sCMA) secure if $\Pr[\mathbf{Exp}^{\text{SU}-\text{sID}-\text{sCMA}}_{\mathcal{A},\text{IBFHS}}(1^\lambda)] \leq \text{negl}(\lambda)$.

5.2 Construction

We use the IBHTDF and homomorphic evaluation given in Sect. 3 and Sect. 4 to construct a leveled IBFHS.

Let $\mathcal{F} = (\mathbf{IBHTDF.Setup}, \mathbf{IBHTDF.Extract}, f, \mathbf{Invert}, \mathbf{IBHTDF.Eval^{in}}, \mathbf{IBHTDF.Eval^{out}})$ be an IBHTDF with identity space $\mathcal{I}$, index space $\mathcal{X}$, input space $\mathcal{U}$, output space $\mathcal{V}$ and some efficiently samplable input distribution $\mathcal{D}_\mathcal{U}$ over $\mathcal{U}$. We construct an IBFHS scheme $\mathcal{S} = (\mathbf{PrmsGen}, \mathbf{Setup}, \mathbf{Extract}, \mathbf{Sign}, \mathbf{SignEval}, \mathbf{Process}, \mathbf{Verify})$ with message space $\mathcal{X}$ as follows.

- $prms \leftarrow \mathbf{PrmsGen}(1^\lambda, 1^\ell)$: Sample $v_i \xleftarrow{\$} \mathcal{V}$, $i \in [N]$ and set public parameters $prms = (v_1, \ldots, v_N)$.

- $(mpk, msk) \leftarrow \mathbf{Setup}(1^\lambda)$: Select$(mpk', msk') \leftarrow \mathbf{IBHTDF.Setup}(1^\lambda)$ and set master-key pair $(mpk = mpk',\ msk = msk')$.
- $(pk_{id}, sk_{id}) \leftarrow \mathbf{Extract}(mpk, msk, id)$: Run $\mathbf{IBHTDF.Extract}(mpk', msk', id)$ to get (pk'_{id}, sk'_{id}) and set $pk_{id} = pk'_{id}, sk_{id} = sk'_{id}$ for $id \in \mathcal{I}$.
- $(\sigma_1, \ldots, \sigma_N) \leftarrow \mathbf{Sign}_{sk_{id}}(prms, x_1, \ldots, x_N)$: Sample $u_i \leftarrow \mathbf{Invert}_{sk'_{id}, x_i}(v_i)$ and set $\sigma_i = u_i, i \in [N]$.
- $\sigma_g = \mathbf{SignEval}_{prms}(g, (x_1, \sigma_1), \ldots, (x_\ell, \sigma_\ell))$: perform deterministic algorithm $\mathbf{IBHTDF.Eval^{in}}\ (g, (x_1, u_1, v_1), \ldots, (x_\ell, u_\ell, v_\ell))$ to get u_g and set $\sigma_g = u_g$.
- $v_g = \mathbf{Process}_{prms}(g)$: Perform $\mathbf{IBHTDF.Eval^{out}}(g, v_1, \ldots, v_\ell)$ and output the result v_g.
- $\mathbf{Verify}_{pk_{id}}(v_g, y, \sigma_g)$: If $f_{pk'_{id}, y}(\sigma_g) = v_g$ accept, else reject.

Security. We now show the SU-sID-sCMA security of the leveled IBFHS.

Lemma 4. *The leveled* IBFHS *scheme $\mathcal{S}$ constructed above is SU-sID-sCMA secure assuming that $\mathcal{F}$ is a leveled selective-identity secure* IBHTDF*.*

Proof. Assume that there exist a PPT adversary $\mathcal{A}$ that wins the security experiment $\mathbf{Exp}^{\mathrm{SU-sID-sCMA}}_{\mathcal{A},\mathrm{IBFHS}}(1^\lambda)$ of IBFHS with non-negligible probability δ. We can construct a PPT reduction $\mathcal{B}$ that breaks the selective-identity security of $\mathcal{F}$.

Let id^* be the identity that $\mathcal{A}$ intends to attack. $\mathcal{B}$ will run the changed algorithms($\mathbf{PrmsGen}^*,\ \mathbf{Setup}^*,\ \mathbf{Extract}^*,\ \mathbf{Sign}^*$).

- $\mathbf{Setup}^*(1^\lambda)$: Run $(mpk', msk') \leftarrow \mathbf{IBHTDF.Setup}^*(1^\lambda)$ and set $mpk = mpk', msk = msk'$.
- $\mathbf{Extract}^*(mpk, msk, id)$: Run $\mathbf{IBHTDF.Extract}^*(mpk, \mathbf{R}, id)$ to get (pk'_{id}, sk'_{id}). When $id \neq id^*$ and set $pk_{id} = pk'_{id}, sk_{id} = sk'_{id}$. However, if $id = id^*$, then the trapdoor disappears and $\mathcal{B}$ can not generate the identity key for id^*.
- $\mathbf{PrmsGen}^*(1^\lambda, 1^N)$: Choose $u_i \leftarrow \mathcal{D}_\mathcal{U}$ and compute $v_i = f_{pk_{id^*}, x_i}(u_i)$. Output $prms = (v_1, \ldots, v_N)$.
- $\mathbf{Sign}^*(x_1, \ldots, x_N)$: Set $\sigma_i = u_i$ and output $(\sigma_1, \ldots, \sigma_N)$.

As the *Distributional Equivalence of Inversion* property underlying **IBHTDF** discussed above, the views of adversary $\mathcal{A}$ between the original experiment and the changed experiment are distinguishable. In particular, the winning probability of $\mathcal{A}$ attacking the changed experiment is at least $\delta - \mathrm{negl}(\lambda)$.

For any PPT adversary $\mathcal{A}$ which can win the changed experiment with non-negligible probability $\delta - \mathrm{negl}(\lambda)$, we now show that there exists a PPT reduction $\mathcal{B}$ can break the security of $\mathcal{F}$ with probability $\delta - \mathrm{negl}(\lambda)$ by access to $\mathcal{A}$.

The reduction $\mathcal{B}$ receives the challenge identity id^* and message data-set $(x_1, \ldots, x_N)$, generates $(mpk, msk, \{\sigma_i = u_i, v_i\}_{i \in [N]})$ and send $(mpk, \{\sigma_i, v_i\}_{i \in [N]})$ to $\mathcal{A}$. If $id \neq id^*$ then $\mathcal{B}$ can respond to any identity-key query for id by msk. But, if $id = id^*$, then the trapdoor disappears, $\mathcal{B}$ doesn't have ability to generate identity-key for id^*.

Assume the adversary $\mathcal{A}$ wins the $\mathbf{Exp}^{\mathrm{SU-sID-sCMA}}_{\mathcal{A},\mathrm{IBFHS}}(1^\lambda)$ that means $\mathcal{A}$ outputs value (g, y', σ'). $g : \mathcal{X}^\ell \to \mathcal{X}$ on $(x_1, \ldots, x_\ell)$ is an admissible function and $\sigma' = u'$. Let $y = g(x_1, \ldots, x_\ell), u_g = \sigma_g = \mathbf{SignEval}_{prms}(g, (x_1, \sigma_1), \ldots, (x_\ell, \sigma_\ell))$, $v_g = \mathbf{Process}_{prms}(g)$. On the one hand, since σ' could verify, $f_{pk_{id^*}, y'}(u') = v_g$ holds. On the other hand, $f_{pk_{id^*}, y}(u_g) = v_g$ must hold because of the correctness of homomorphic computation. Then we have $u_g \neq u' \in \mathcal{U}$ and $y, y' \in \mathcal{X}$ satisfying $f_{pk_{id^*}, y}(u_g) = f_{pk_{id^*}, y'}(u')$, that allow $\mathcal{B}$ break the $\mathbf{Exp}^{\mathrm{sID}}_{\mathcal{A},\mathrm{IBHTDF}}(1^\lambda)$ security of $\mathcal{F}$ with probability $\delta-\mathrm{negl}(\lambda)$ whenever $\mathcal{A}$ wins the changed experiment with probability $\delta-\mathrm{negl}(\lambda)$. This complete the proof of this lemma.

6 Conclusions

In this work, we construct a new leveled strongly-unforgeable IBFHS scheme which is based on our new IBHTDF. The maximum noise level of addition gate and multiplication gate are exactly the same in our IBHTDF. That means that the maximum noise level of our IBHTDF is optimal. It remains open to decrease the leveled aspect and ideally come up with a signature scheme where there is no priori bound on the depth of the circuits that can be efficiently evaluated with short public parameters. What's more, the existence of any other schemes, e.g. homomorphic encryption or ABE, could use our trapdoor generation technique is still a puzzle.

References

1. Agrawal, S., Boneh, D., Boyen, X.: Efficient lattice (H)IBE in the standard model. In: Gilbert, H. (ed.) EUROCRYPT 2010. LNCS, vol. 6110, pp. 553–572. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13190-5_28
2. Brakerski, Z., Vaikuntanathan, V.: Lattice-based FHE as secure as PKE. In: ITCS, pp. 1–12 (2014)
3. Boyen, X., Fan, X., Shi, E.: Adaptively secure fully gomomorphic signatures based on lattices. Cryptology ePrint Archive, Report 2014/916. http://eprint.iacr.org/2014/916
4. Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: STOC, pp. 197–206. ACM, New York (2008)
5. Gentry, C.: Fully homomorphic encryption using ideal lattices. In: STOC 2009, pp. 169–178. ACM (2009)
6. Gorbunov, S., Vaikuntanathan, V., Wichs, D.: Leveled fully homomorphic signatures from standard lattices. In: STOC 2015, pp. 469–477. ACM (2015)
7. Micciancio, D., Peikert, C.: Hardness of SIS and LWE with small parameters. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8042, pp. 21–39. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40041-4_2
8. Micciancio, D., Peikert, C.: Trapdoors for lattices: simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 700–718. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-29011-4_41
9. Xie, X., Xue, R.: Bounded Fully Homomorphic Signature Schemes. IACR Cryptology ePrint Archive, p. 420 (2014)

10. Wang, F., Wang, K., Li, B., Gao, Y.: Leveled strongly-unforgeable identity-based fully homomorphic signatures. In: Lopez, J., Mitchell, C.J. (eds.) ISC 2015. LNCS, vol. 9290, pp. 42–60. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-23318-5_3
11. Micciancio, D., Regev, O.: Worst-case to average-case reductions based on gaussian measures. SIAM J. Comput. **37**, 267–302 (2007)