Ideal Multilinear Maps Based on Ideal Lattices

Gu Chunsheng

School of Computer Engineering, Jiangsu University of Technology, Changzhou 213001, China
E-mail: chunsheng_gu@163.com

March 23, 2015

Abstract. Cryptographic multilinear maps have found many applications, such as multipartite Diffie-Hellman key exchange, general software obfuscation. However, currently only three constructions are known, and are “noisy” and bounded to polynomial degree. In this paper, we describe constructions of ideal multilinear maps using ideal lattices, which support arbitrary multilinearity levels. The security of our construction depends on hardness assumption over ideal lattices. Moreover, we describe one-round multipartite Diffie-Hellman key exchange protocols by using our construction.

Keywords. Ideal multilinear maps, Ideal lattices, Diffie-Hellman key exchange, Zeroizing attack

1 Introduction

Constructing multilinear maps have been a long-standing open problem since 2003. Many applications on bilinear maps, such as [SOK00, Jou00, BF01, Sma03], et al., inspired the study of cryptographical multilinear maps [BS03, RS09, PTT10, Rot13]. Boneh and Silverberg [BS03] first introduced the notion of multilinear maps, which are an extension of bilinear maps. However, they suspected that such maps come from the realm of algebraic geometry.

Garg, Gentry, and Halevi recently described the first plausible construction of multilinear maps that use ideal lattices [GGH13]. Their multilinear maps, whose encodings were randomized with noise and bounded with a fixed maximum degree, were different from the ideal multilinear maps of Boneh and Silverberg. Construction security depends on the new hardness assumptions of GCDH/GDDH, which are provided an extensive cryptanalysis in [GGH13]. To reduce the public parameter size of GGH, Langlois, Stehlé, and Steinfeld [LSS14] improved the security analysis of the GGH construction re-randomization process and decreased the public parameter bit size for the GGH scheme from $O(k^3\lambda^5\log(k\lambda))$ to $O(k^3\lambda\log^2(k\lambda))$ in GGHLite, with respect to security parameter λ and multilinearity parameter k. Although the length of public parameters of the GGHLite is asymptotically close to optimal, a large hidden constant is found in $O(k^3\lambda\log^2(k\lambda))$.

Following GGH’s idea, Coron, Lepoint, and Tibouchi [CLT13] described a new and relatively practical construction following the [GGH13] method. Their construction works over integers instead of ideal lattices, and multilinear maps are implemented over integers using heuristic optimization techniques. However, the CLT multilinear maps have been broken by Cheon et al. [CHL+14] using level-1 encodings of zero. To fix the construction [CLT13], Boneh, Wu, and Zimmerman [BWZ14] and Garg, Gentry, Halevi, and Zhandry [GGHZ14] proposed two independent approaches to avoid zeroizing attack, however Coron，Lepoint and Tibouchi [CTL14] show that two fixes can be defeated using extensions of the [CHL+14].

To improve the security of previous constructions, Hiromasa, Abe and Okamoto [HAO14] constructed new multilinear maps based on GSW’s fully homomorphic encryption. The security of their construction is not reduced to LWE, although the security of GSW’s fully homomorphic encryption is reduced to LWE. Basing same idea, Gentry，Gorbunov and Halevi [GGH14] described graph-induced multilinear maps from lattices. Their construction encodes LWE samples on short square matrix with higher dimension, however, the security of their construction is not reduced to LWE or hard assumption of arbitrary other classic problems.

To avoid zeroizing attack in the GGH construction, Gu [Gu15] described a construction of multilinear maps without encoding of zero by designing new zero-testing parameters. Recently, Coron, Lepoint, and Tibouchi fixed the construction in [CTL13] by also modifying zero-testing parameters.

However, all current constructions follow the framework of the GGH construction, whose levels are in advance fixed and encodings have noisy. In this paper, we will describe a construction of ideal multilinear maps from ideal lattices using the methods in [Gu15, CLT15].

Our Results. Our main contribution is presenting a construction of ideal multilinear maps that use ideal lattices. Construction security depends on a new hardness assumption. Our construction works in a polynomial ring $R=\mathbb{Z}[x]/(x^n+1)$, where n is a positive integer. Given secret ring elements $\mathbf{f}_j,\mathbf{g}_j\in R, j=1,...,m$, we denote $\mathbf{f}=\prod_{j=1}^{m}\mathbf{f}_j$. A level-$1$ encoding of level-0 element $\mathbf{a}\in R$ is $\mathbf{c}=(\mathbf{a}\cdot\mathbf{g})\bmod\mathbf{f}$, where $\mathbf{g}=\mathbf{g}_j\bmod\mathbf{f}_j$, $\mathbf{a}=\mathbf{a}_j\bmod\mathbf{f}_j$, $j=1,...,m$. If only given $\mathbf{a}$ and $\mathbf{c}$, then one can compute $\mathbf{g}=(\mathbf{c}/\mathbf{a})\bmod\mathbf{f}$. So, we provide the multiplier $\mathbf{q}=\mathbf{q}_0\cdot\mathbf{f}$ of $\mathbf{f}$ in the public parameters to avoid this simple attack. Now, we transform the encoding $\mathbf{c}$ to a new encoding $\mathbf{u}=(\mathbf{c}+\mathbf{r}\cdot\mathbf{f})\bmod\mathbf{q}$. To decide whether or not $\mathbf{u}$ is an encoding of zero, we provide a zero-testing parameter $\mathbf{p}_{zt}=(\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q$ in the public parameters. If the norm of $\left[\mathbf{p}_{zt}\cdot\mathbf{u}\right]_q$ is small, then $\mathbf{u}$ is the encoding of zero; otherwise, it is the encoding of non-zero. This defines an arbitrary degree multilinear map.

Our second contribution is presenting a variant of ideal construction to avoid zeroizing attack in the above construction. This is because $\mathbf{q}$ is an encoding of zero. To support arbitrary multilinearity levels, we must provide this encoding of zero. So in the variant, we take a large enough multiplier $\mathbf{q}_0$ so that a non-reduced quantity over the modulo q cannot be directly obtained multiplying $\mathbf{q}$ by $\mathbf{p}_{zt}$. In the public parameters, we provide a list of non-zero encodings and its corresponding zero-testing parameters to gradually reduce encoding. We have used this method of constructing new zero-testing parameters in [Gu15] to improve the GGH construction [GGH13].

Our third contribution is presenting a one-round multipartite Diffie-Hellman key exchange protocol, which supports arbitrary multilinearity levels. The integer modulo in our construction does not increase by multilinearity levels. Thus, our ideal multilinear maps using ideal lattices are practical.

The remainder of this paper is organized as follows: some preliminaries are recalled in Section 2, construction of ideal multilinear maps that use ideal lattices is described in Section 3, a variant of our ideal construction is presented in Section 4, and a symmetric ideal variant is presented in Section 5. Finally, one-round multipartite Diffie-Hellman key exchange protocol is constructed in Section 6.

2 Preliminaries

2.1 Notations

We denote $\mathbb{Z},\mathbb{Q},\mathbb{R}$ the integer ring, rational number field and real number field. Vectors and matrices are denoted in bold. We denote $[\![k]\!]=\{1,2,\cdots,k\}$ for $k\in\mathbb{N}$. We take n as

a power of two, the polynomial ring $R = \mathbb{Z}[x]/ < x^n + 1 >$ and polynomial field $\mathbb{k} = \mathbb{Q}[x]/ < x^n + 1 >$. For $\mathbf{a} \in R$, $\|\mathbf{a}\|_\infty$ ($\|\mathbf{a}\|$ for short) denotes the infinity norm of the vector corresponding to $\mathbf{a}$.
In this paper, we use the absolute minimum residual system, that is $[a]_q = a \bmod q \in (-q/2, q/2]$. Similarly, notation $[\mathbf{a}]_q$ denotes each entry (or each coefficient) $a_i \in (-p/2, p/2]$.

2.2 Lattices and Ideal Lattices

An *n*-dimension full-rank lattice $L \subset \mathbb{R}^n$ is the set of all integer linear combinations $\sum_{i=1}^n y_i \mathbf{b}_i$ of *n* linearly independent vectors $\mathbf{b}_i \in \mathbb{R}^n$. If we arrange the vectors $\mathbf{b}_i$ as the columns of matrix $B \in \mathbb{R}^{n \times n}$, then $L = \left\{\mathbf{By} : \mathbf{y} \in \mathbb{Z}^n\right\}$. We say that $\mathbf{B}$ spans L if $\mathbf{B}$ is a basis for L. For any lattice basis, we define $P(\mathbf{B}) = \{\mathbf{Bz} \mid \mathbf{z} \in \mathbb{R}^n, \forall i : -1/2 \le z_i < 1/2\}$. Let $\det(\mathbf{B})$ denote the determinant of the matrix $\mathbf{B}$.

Given $\mathbf{a}, \mathbf{g} \in R$, we let the principal ideal $I = < \mathbf{g} >$ with the R_1-basis $Rot(\mathbf{g}) = (\mathbf{g}, x_2 \cdot \mathbf{g}, ..., x_2^{n-1} \cdot \mathbf{g})$ and $[\mathbf{a}]_\mathbf{g}$ denote the modulo reduction of $I = < \mathbf{g} >$, namely, $[\mathbf{a}]_\mathbf{g} \in P(Rot(\mathbf{g}))$ and $(\mathbf{a} - [\mathbf{a}]_\mathbf{g}) \in L(Rot(\mathbf{g}))$.

Given $\mathbf{c} \in \mathbb{R}^n$, $\sigma > 0$, we define $D_{L,\sigma,\mathbf{c}} = \rho_{\sigma,\mathbf{c}}(\mathbf{x}) / \rho_{\sigma,\mathbf{c}}(L)$ the Gaussian distribution of a lattice L, where $\mathbf{x} \in L$, $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp(-\pi \|\mathbf{x} - \mathbf{c}\|^2 / \sigma^2)$, $\rho_{\sigma,\mathbf{c}}(L) = \sum_{x \in L} \rho_{\sigma,\mathbf{c}}(\mathbf{x})$. In the following, we will write $D_{\mathbb{Z}^n,\sigma,0}$ as $D_{\mathbb{Z}^n,\sigma}$. We denote a Gaussian sample as $\mathbf{x} \leftarrow D_{L,\sigma}$ (or $\mathbf{d} \leftarrow D_{I,\sigma}$) over the lattice L (or ideal lattice I).

2.3 Multilinear Maps

Definition 2.1 (Multilinear Map [BS03]). For $k+1$ cyclic groups $G_1, ...G_k, G_T$ of the same order p, a k-multilinear map $e : G_1 \times \cdots \times G_k \to G_T$ has the following properties:

(1)Elements $\left\{g_j \in G_j\right\}_{j=1,...,k}$, index $j \in [\![k]\!]$, and integer $a \in Z_p$ hold that

$$e(g_1, \cdots, a \cdot g_j, \cdots, g_k) = a \cdot e(g_1, \cdots, g_k).$$

(2)Map e is non-degenerate in the following sense: if elements $\left\{g_j \in G_j\right\}_{j=1,...,k}$ are generators of their respective groups, then $e(g_1, \cdots, g_k)$ is a generator of G_T.

Definition 2.2 (k-Graded Encoding System [GGH13]). A k-graded encoding system over R is a set system of $S = \left\{S_j^{(\alpha)} \subset R : \alpha \in R, j \in [\![k]\!]\right\}$ with the following properties:

(1) For every index $j \in [\![k]\!]$, the sets $\left\{S_j^{(\alpha)} : \alpha \in R\right\}$ are disjoint.

(2) Binary operations '$+$' and '$-$' exist, such that every α_1, α_2, every index $j \in [\![k]\!]$, and every $u_1 \in S_j^{(\alpha_1)}$ and $u_2 \in S_j^{(\alpha_2)}$ hold that $u_1 + u_2 \in S_j^{(\alpha_1+\alpha_2)}$ and $u_1 - u_2 \in S_j^{(\alpha_1-\alpha_2)}$, where $\alpha_1 + \alpha_2$ and $\alpha_1 - \alpha_2$ are the addition and subtraction operations in R

respectively.

(3) Binary operation ‘×’ exists, such that every α_1, α_2, every index $j_1, j_2 \in [\![k]\!]$ with $j_1 + j_2 \le k$, and every $u_1 \in S_{j_1}^{(\alpha_1)}$ and $u_2 \in S_{j_2}^{(\alpha_2)}$ hold that $u_1 \times u_2 \in S_{j_1+j_2}^{(\alpha_1 \times \alpha_2)}$, where $\alpha_1 \times \alpha_2$ is the multiplication operation in R and $j_1 + j_2$ is the integer addition.

3 Ideal Multilinear Maps

In this section, we first construct symmetric multilinear maps over ideal lattices. Then we show the correctness of our construction. Next, we show the security of our construction. Finally, we give known cryptanalysis for our construction.

3.1 Construction

Setting the parameters. Because our construction uses the GGH construction as the basic component, our parameter setting is set as that of GGH to conveniently describe and compare. Let λ be the security parameter, n the dimension of elements of R. Concrete parameters are set as $\sigma = \sqrt{\lambda n}$, $\sigma' = \lambda n^{1.5}$, $q \ge 2^{\eta} n^{O(1)}$, $n = O(\lambda^2)$, $m = 2$, $l = O(\lambda^2)$, $\tau = O(n^2)$.

Instance generation: $(\mathrm{par}_1) \leftarrow \mathrm{InstGen}_1(1^{\lambda})$.

(1) Choose a large enough prime q

(2) Sample $\mathbf{a}_{i,j}, \mathbf{f}_j, \mathbf{g}_j, \mathbf{h}_j \leftarrow D_{\mathbb{Z}^n, \sigma}$, $\mathbf{s}_i, \mathbf{t}_i \leftarrow D_{\mathbb{Z}^n, \sigma'}$, $i \in [\![\tau]\!]$, $j \in [\![m]\!]$ and $\mathbf{p}_0 \leftarrow D_{\mathbb{Z}^n, \sigma'}$ such that all ideal lattices $\mathbf{f}_j$ ’s are coprime, $\mathbf{f}_j^{-1} \in \mathbb{k}$ and $\left\| \mathbf{f}_j^{-1} \right\| \le l$.

(3) Compute $\mathbf{f} = \prod_{j=1}^{m} \mathbf{f}_j$ and $\mathbf{q} = \mathbf{p}_0 \cdot \mathbf{f}$.

(4) Compute $\mathbf{d}_i$, $\mathbf{c}_i$, $i \in [\![\tau]\!]$ over the modulo $\mathbf{f}$ such that $\mathbf{d}_i = \mathbf{a}_{i,j} \bmod \mathbf{f}_j$, $\mathbf{c}_i = (\mathbf{a}_{i,j} \cdot \mathbf{g}_j) \bmod \mathbf{f}_j$, $j \in [\![m]\!]$.

(5) Set $\mathbf{x}_i = (\mathbf{d}_i + \mathbf{s}_i \cdot \mathbf{f}) \bmod \mathbf{q}$ and $\mathbf{y}_i = (\mathbf{c}_i + \mathbf{t}_i \cdot \mathbf{f}) \bmod \mathbf{q}, i \in [\![\tau]\!]$.

(6) Set $\mathbf{p}_{zt} = (\sum_{j=1}^{m} \mathbf{h}_j \cdot (\mathbf{f}_j^{-1} \bmod q)) \bmod q$.

(7) Output public parameters $\mathrm{par}_1 = \left\{ q, \mathbf{q}, \left\{ \mathbf{x}_i, \mathbf{y}_i \right\}_{i \in [\![\tau]\!]}, \mathbf{p}_{zt} \right\}$.

Generating level-k random encodings: $\mathbf{u} \leftarrow \mathrm{Enc}_1(\mathrm{par}_1, k, \{\mathbf{w}_i\}_{i \in [\![\tau]\!]})$.

Choose $\mathbf{w}_i \leftarrow D_{\mathbb{Z}^n, \sigma'}$, $i \in [\![\tau]\!]$, generate a level- 0 encoding $\mathbf{d} = \sum_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{x}_i)^k \bmod \mathbf{q}$ and a level-k encoding $\mathbf{u} = \sum_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{y}_i)^k \bmod \mathbf{q}$.

Adding encodings: $\mathbf{u} \leftarrow \mathrm{Add}_1(\mathrm{par}_1, k, \mathbf{u}_1, \cdots, \mathbf{u}_s)$.

Given s level-k encodings $\mathbf{u}_t$, $t \in [\![s]\!]$, their sum $\mathbf{u} = \sum_{t=1}^{s} \mathbf{u}_t \bmod \mathbf{q}$ is a level-k encoding.

Multiplying encodings: $\mathbf{u} \leftarrow \mathrm{Mul}_1(\mathrm{par}_1, 1, \mathbf{u}_1, \cdots, \mathbf{u}_k)$.

Given k level-1 encodings $\mathbf{u}_t$, $t \in [\![k]\!]$, their product $\mathbf{u} = \prod_{t=1}^{k} \mathbf{u}_t \bmod \mathbf{q}$ is a level-k encoding.

Zero Testing: $\mathrm{isZero}_1(\mathrm{par}_1, \mathbf{u})$ **.**

To determine whether level-k encoding $\mathbf{u}$ is an encoding of zero, $\mathbf{v} = [\mathbf{p}_{zt} \cdot \mathbf{u}]_q$ is computed in R_q and checked whether $\|\mathbf{v}\|$ is short:

$$\text{isZero}_1(\text{par}_1,\mathbf{u}) = \begin{cases} 1 & \text{if } \left\|[\mathbf{p}_{zt}\cdot\mathbf{u}]_q\right\| < q/2^{\eta} \\ 0 & \text{otherwise} \end{cases}.$$

Extract: $sk \leftarrow \text{Ext}(\text{par}_1,\mathbf{u})$.

Given a level-k encoding $\mathbf{u}$, $\text{Ext}_1(\text{par}_1,\mathbf{u}) = \text{Extract}_s(\text{msbs}_{\eta}([\mathbf{p}_{zt}\cdot\mathbf{u}]_q))$.

In this paper, we omit the seed s and concrete extraction algorithm Extract.

Remark 3.1 When $m>1$, one requires to solve $(\mathbf{f}/\mathbf{f}_j)^{-1}\bmod \mathbf{f}_j$. The inverse operation over the modulo $\mathbf{f}_j$ needs to cost expensive time. One can set $m=1$ and a large dimension n to avoid this costly inverse computation.

3.2 Correctness

Lemma 3.2 If the ideal lattices $\mathbf{f},\mathbf{g}$ over R are co-prime, then there exists a PPT algorithm which solves $\mathbf{s},\mathbf{t}$ such that $\mathbf{sf}+\mathbf{tg}=1$.

Proof. Using LLL algorithm, we can obtain a basis $\mathbf{E}$ generated by $\mathbf{f},\mathbf{g}$ and two matrices $\mathbf{U}_1,\mathbf{U}_2$ such that $\mathbf{U}_1(Rot(\mathbf{f}))^{\mathrm{T}} + \mathbf{U}_2(Rot(\mathbf{g}))^{\mathrm{T}} = \mathbf{E}$. Since $\mathbf{f},\mathbf{g}$ are co-prime, $|\det(\mathbf{E})|=1$. Namely, $\mathbf{E}$ is a unimodular matrix. So, $\mathbf{E}^{-1}$ is also unimodular and in particular $\mathbf{E}^{-1}\in\mathbb{Z}^{n\times n}$. So, $\mathbf{E}^{-1}\mathbf{U}_1(Rot(\mathbf{f}))^{\mathrm{T}} + \mathbf{E}^{-1}\mathbf{U}_2(Rot(\mathbf{g}))^{\mathrm{T}} = \mathbf{I}$ and $Rot(\mathbf{f})(\mathbf{E}^{-1}\mathbf{U}_1)^{\mathrm{T}} + Rot(\mathbf{g})(\mathbf{E}^{-1}\mathbf{U}_2)^{\mathrm{T}} = \mathbf{I}$. Let $\mathbf{s}$, $\mathbf{t}$ be the first column of $(\mathbf{E}^{-1}\mathbf{U}_1)^{\mathrm{T}}$, $(\mathbf{E}^{-1}\mathbf{U}_2)^{\mathrm{T}}$, respectively. It is easy to verify that $\mathbf{sf}+\mathbf{tg}=1$.

Lemma 3.3 The instance generation $\text{InstGen}_1(1^{\lambda})$ is a probabilistic polynomial time algorithm.

Proof. One can efficiently generate a prime q. By [GGH13], one can sample $\mathbf{a}_{i,j},\mathbf{f}_j,\mathbf{g}_j,\mathbf{h}_j$ such that $\mathbf{f}_j^{-1}\in\mathbb{k}$ and $\|\mathbf{f}_j^{-1}\|\le l$ with high probability, and compute $\mathbf{f}$ and $\mathbf{q}$.

According to Chinese remainder theorem, we have

$$\mathbf{d}_i = (\textstyle\sum_{j=1}^{m}\mathbf{a}_{i,j}\cdot(\mathbf{f}/\mathbf{f}_j)\cdot(\mathbf{f}/\mathbf{f}_j)^{-1}\bmod \mathbf{f}_j)\bmod \mathbf{f}.$$

By Lemma 3.2, we know that $(\mathbf{f}/\mathbf{f}_j)^{-1}\bmod \mathbf{f}_j$ can be solved in polynomial time. Namely, one can compute $\mathbf{d}_i$ in polynomial time. Similarly, one can get $\mathbf{c}_i$ in polynomial time.

It is easy to see that $\mathbf{x}_i,\mathbf{y}_i,\mathbf{p}_{zt}$ can be generated in polynomial time. ■

Lemma 3.4 $\mathbf{u}\leftarrow\text{Enc}_1(\text{par}_1,k,\mathbf{d})$ is a level-k encoding of $\mathbf{d}$.

Proof. Since $\mathbf{d}=\sum_{i=1}^{\tau}\mathbf{w}_i\cdot(\mathbf{x}_i)^k \bmod \mathbf{q}$, we have

$$\begin{aligned} &\mathbf{d} \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{x}_i)^k \bmod \mathbf{q}) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot ((\mathbf{d}_i + \mathbf{s}_i \cdot \mathbf{f}) \bmod \mathbf{q})^k \bmod \mathbf{q}) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{d}_i + \mathbf{s}_i \cdot \mathbf{f})^k \bmod \mathbf{q}) \bmod \mathbf{f}_j \qquad . \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{d}_i + \mathbf{s}_i \cdot \mathbf{f})^k) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{d}_i \bmod \mathbf{f}_j)^k) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{a}_{i,j})^k) \bmod \mathbf{f}_j \end{aligned}$$

Since $\mathbf{u} = \sum_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{y}_i)^k \bmod \mathbf{q}$, we have

$$\begin{aligned} &\mathbf{u} \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{y}_i)^k \bmod \mathbf{q}) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot ((\mathbf{c}_i + \mathbf{t}_i \cdot \mathbf{f}) \bmod \mathbf{q})^k \bmod \mathbf{q}) \bmod \mathbf{f}_j \qquad . \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot ((\mathbf{c}_i + \mathbf{t}_i \cdot \mathbf{f}) \bmod \mathbf{f}_j)^k) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{a}_{i,j}\mathbf{g}_j)^k) \bmod \mathbf{f}_j \\ &= (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{a}_{i,j})^k) \cdot (\mathbf{g}_j)^k \bmod \mathbf{f}_j \end{aligned}$$

So, $\mathbf{u}$ is a level-k encoding of $\mathbf{d}$. ■

Lemma 3.5 $\mathbf{u} \leftarrow \mathrm{Add}_1(\mathrm{par}_1, k, \mathbf{u}_1, \cdots, \mathbf{u}_s)$ is a level-k encoding.

Proof. Using modulo operation, it is easy to verify that $\mathbf{u}$ is a level-k encoding. ■

Lemma 3.6 $\mathbf{u} \leftarrow \mathrm{Mul}_1(\mathrm{par}_1, 1, \mathbf{u}_1, \cdots, \mathbf{u}_k)$ is a level-k encoding.

Proof. Since $\mathbf{u}_t, t \in [\![k]\!]$ are level-1 encodings, we have $\mathbf{u}_t \bmod \mathbf{f}_j = \mathbf{u}_{t,j}\mathbf{g}_j$. Then, we have

$$\begin{aligned} &\mathbf{u} \bmod \mathbf{f}_j \\ &= (\prod\nolimits_{t=1}^{k} \mathbf{u}_t \bmod \mathbf{q}) \bmod \mathbf{f}_j \\ &= (\prod\nolimits_{t=1}^{k} \mathbf{u}_t \bmod \mathbf{f}_j) \bmod \mathbf{f}_j \\ &= (\prod\nolimits_{t=1}^{k} (\mathbf{u}_t \bmod \mathbf{f}_j)) \bmod \mathbf{f}_j \\ &= (\prod\nolimits_{t=1}^{k} \mathbf{u}_{t,j}\mathbf{g}_j) \bmod \mathbf{f}_j \\ &= (\prod\nolimits_{t=1}^{k} \mathbf{u}_{t,j})(\mathbf{g}_j)^k \bmod \mathbf{f}_j \end{aligned}$$

So, $\mathbf{u}$ is a level-k encoding. ■

Lemma 3.7 For an arbitrary integer $k > 0$, the zero-testing algorithm $\mathrm{isZero}_1(\mathrm{par}_1, \mathbf{u})$ correctly determines whether a level-k encoding $\mathbf{u}$ is an encoding of zero.

Proof. Given an arbitrary level-k encoding $\mathbf{u}$, we have $\mathbf{u} = \mathbf{d} + \mathbf{r} \cdot \mathbf{f}$ and $\|\mathbf{u}\| < \|\mathbf{q}\|$ with $\|\mathbf{d}\| < \|\mathbf{f}\|$.

(1) If $\mathbf{u}$ is an encoding of zero, then $\mathbf{u} \bmod \mathbf{f}_j = 0, j \in [\![m]\!]$. Since $\mathbf{f}_j, j \in [\![m]\!]$ are co-prime, $\mathbf{u} \bmod \mathbf{f} = 0$. That is, $\mathbf{d} \bmod \mathbf{f} = 0$ and $\mathbf{d} = 0$ according to $\|\mathbf{d}\| < \|\mathbf{f}\|$. So, we have

$$\begin{aligned}
\mathbf{v} &= \left\|\left[\mathbf{p}_{zt}\cdot\mathbf{u}\right]_q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q\cdot\mathbf{u})\bmod q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{u}\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q)\bmod q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}\cdot\mathbf{f}\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q)\bmod q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}\cdot\mathbf{f}/\mathbf{f}_j)\bmod q)\bmod q\right\| \\
&= \left\|(\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}\cdot\mathbf{f}/\mathbf{f}_j)\bmod q\right\| \\
&\le \left\|\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}\cdot\mathbf{f}/\mathbf{f}_j\right\| \\
&\le \textstyle\sum_{j=1}^{m}\left\|\mathbf{h}_j\cdot\mathbf{r}\cdot\mathbf{f}/\mathbf{f}_j\right\| \\
&\le \textstyle\sum_{j=1}^{m}\left\|\mathbf{h}_j\right\|\cdot\left\|\mathbf{r}\right\|\cdot\left\|\mathbf{f}/\mathbf{f}_j\right\| \\
&\le q/2^{\eta}
\end{aligned} .$$

(2) If $\mathbf{u}$ is not an encoding of zero, then $\mathbf{u}\bmod\mathbf{f}\neq 0$. That is, $\exists j\in[\![m]\!],\mathbf{d}\bmod\mathbf{f}_j\neq 0$. So,

$$\begin{aligned}
\mathbf{v} &= \left\|\left[\mathbf{p}_{zt}\cdot\mathbf{u}\right]_q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q\cdot\mathbf{u})\bmod q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{u}\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q)\bmod q\right\| \\
&= \left\|((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{d}+\mathbf{r}\cdot\mathbf{f})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q)\bmod q\right\| \\
&= \left\|(\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}\cdot\mathbf{f}/\mathbf{f}_j)+(\textstyle\sum_{j=1}^{m}(\mathbf{h}_j\cdot\mathbf{d})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q\right\| \\
&\ge \left\|(\textstyle\sum_{j=1}^{m}(\mathbf{h}_j\cdot\mathbf{d})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q\right\|-\textstyle\sum_{j=1}^{m}\left\|\mathbf{h}_j\right\|\cdot\left\|\mathbf{r}\right\|\cdot\left\|\mathbf{f}/\mathbf{f}_j\right\| \\
&\ge q^{1-\varepsilon}-q/2^{\eta} \\
&\ge q^{1-\varepsilon'}
\end{aligned}$$

So, $\mathrm{isZero}_1(\mathrm{par}_1,\mathbf{u})$ correctly decides the encoding of $\mathbf{u}$. ■

Lemma 3.8 If two level-k encodings $\mathbf{u}_1,\mathbf{u}_2$ encode same level-0 element, namely $\mathbf{u}_1=\mathbf{u}_2=\mathbf{a}_j(\mathbf{g}_j)^k\bmod\mathbf{f}_j,\forall j\in[\![m]\!]$, then $\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{u}_1)=\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{u}_2)$.

Proof. Since $\mathbf{u}_1=\mathbf{u}_2=\mathbf{a}_j(\mathbf{g}_j)^k\bmod\mathbf{f}_j,\forall j\in[\![m]\!]$ and $\mathbf{f}_j, j\in[\![m]\!]$ are co-prime, we get $\mathbf{u}_1=\mathbf{d}+\mathbf{r}_1\cdot\mathbf{f},\mathbf{u}_2=\mathbf{d}+\mathbf{r}_2\cdot\mathbf{f}$. So, we have

$$\begin{aligned}
\left[\mathbf{p}_{zt}\cdot\mathbf{u}_1\right]_q &= ((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{d}+\mathbf{r}_1\cdot\mathbf{f})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q)\bmod q \\
&= (\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}_1\cdot\mathbf{f}/\mathbf{f}_j)+(\textstyle\sum_{j=1}^{m}(\mathbf{h}_j\cdot\mathbf{d})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q \\
\left[\mathbf{p}_{zt}\cdot\mathbf{u}_2\right]_q &= ((\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{d}+\mathbf{r}_2\cdot\mathbf{f})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q)\bmod q \\
&= (\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}_2\cdot\mathbf{f}/\mathbf{f}_j)+(\textstyle\sum_{j=1}^{m}(\mathbf{h}_j\cdot\mathbf{d})\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q
\end{aligned} .$$

By Lemma 3.7, we have that $(\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}_1\cdot\mathbf{f}/\mathbf{f}_j)$, $(\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{r}_2\cdot\mathbf{f}/\mathbf{f}_j)$ are less than $q/2^{\eta}$, respectively. Hence, the most significant bits of $[\mathbf{p}_{zt}\cdot\mathbf{u}_1]_q$ and $[\mathbf{p}_{zt}\cdot\mathbf{u}_2]_q$ are all decided by the second term $(\sum_{j=1}^{m}(\mathbf{h}_j\cdot\mathbf{d})\cdot(\mathbf{f}_j^{-1}\bmod q))$. That is, $\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{u}_1)=\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{u}_2)$. ■

3.3 Security

The security of our constructions depends on new hardness assumptions, and seems to rely on hardness to solve shortest generator problems for principal ideal lattices. However, at present we do not know how to reduce the security of our construction to the shortest principal ideal generator problems.

Hardness assumptions for multilinear maps in [GGH13] are modeled as discrete logarithms and DDH assumptions in multilinear groups. Generating a level-k encoding of the product or distinguishing the product from random elements is unfeasible given the public parameters and $k+1$ level-1 encodings of random elements.

Garg, Gentry, and Halevi introduced the definition of GCDH/GDDH in [GGH13] to describe the hardness assumption of the GGH construction. Langlois, Stehlé, and Steinfeld extended the GCDH/GDDH to the ext-GCDH/ext-GDDH in [LSS14] to prove the security of the GGHLite scheme.

In the following, we adapt the definition of ext-GCDH/ext-GDDH in [LSS14] to our constructions. Consider the following process:

(1) $(\mathrm{par}_1)\leftarrow\mathrm{InstGen}_1(1^{\lambda})$.

(2) Choose an arbitrary positive integer k.

(3) For $t=0$ to k:

Sample $\mathbf{w}_{t,i}\leftarrow D_{\mathbb{Z}^n,\sigma'}, i\in[\![\tau]\!]$

Generate level-1 encoding of $\mathbf{d}_t=(\sum_{i=1}^{\tau}\mathbf{w}_{t,i}\cdot\mathbf{x}_i)\bmod\mathbf{q}$:

$\mathbf{u}_t=(\sum_{i=1}^{\tau}\mathbf{w}_{t,i}\cdot\mathbf{y}_i)\bmod\mathbf{q}$.

(4) Sample $\mathbf{r}_{0,i}\leftarrow D_{\mathbb{Z}^n,\sigma'}, i\in[\![\tau]\!]$ and generate $\mathbf{r}_0=(\sum_{i=1}^{\tau}\mathbf{r}_{0,i}\cdot\mathbf{x}_i)\bmod\mathbf{q}$.

(5) Compute $\mathbf{u}^*=\prod_{t=1}^{k}\mathbf{u}_t\bmod\mathbf{q}$, $\mathbf{u}=(\mathbf{d}_0\cdot\mathbf{u}^*)\bmod\mathbf{q}$ and $\mathbf{u}'=\mathbf{r}_0\cdot\mathbf{u}^*\bmod\mathbf{q}$.

(6) Set $\mathbf{v}_C=\mathbf{v}_D=\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{u})$.

(7) Set $\mathbf{v}_R=\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{u}')$.

Definition 3.8 (ext-GCDH/ext-GDDH). The extraction k-graded CDH problem (ext-GCDH) is, on input $\{\mathrm{par}_1,\mathbf{u}_0,\cdots,\mathbf{u}_k\}$, to output an extraction encoding $\mathbf{w}\in R_p$, such that $\mathrm{Ext}_1(\mathrm{par}_1,\mathbf{w})=\mathbf{v}_C$. The extraction k-graded DDH problem (ext-GDDH) distinguishes between $\mathbf{v}_D$ and $\mathbf{v}_R$, that is, between the distributions $D_{GDDH}=\{\mathrm{par}_1,\mathbf{u}_0,\cdots,\mathbf{u}_k,\mathbf{v}_D\}$ and $D_{RAND}=\{\mathrm{par}_1,\mathbf{u}_0,\cdots,\mathbf{u}_k,\mathbf{v}_R\}$.

As in [GGH13], our construction security depends on new assumptions that are unlikely to be reducible to more classical assumptions. We assume the ideal-GCDH/ ideal-GDDH is hard in our scheme.

3.4 Cryptanalysis

In this section, we will describe known attacks of our above construction. In the following section, we will present a variant construction to thwart this attack.

3.3.1 Average Attack

Since $\mathbf{q}=\mathbf{p}_0\cdot\mathbf{f}$ is an encoding of zero in our construction, $\left[\mathbf{q}\cdot\mathbf{p}_{zt}\right]_q$ is not reduced modulo q. So, the following quantities are easily computed from public parameters through algebraic transformation.

$$\begin{aligned}\mathbf{u}&=\left[\mathbf{q}\cdot\mathbf{p}_{zt}\right]_q\\&=(\mathbf{p}_0\cdot\mathbf{f})(\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot(\mathbf{f}_j^{-1}\bmod q))\bmod q\\&=(\mathbf{p}_0\cdot\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{f}/\mathbf{f}_j)\bmod q\\&=\mathbf{p}_0\cdot\textstyle\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{f}/\mathbf{f}_j\\&=\mathbf{p}_0\cdot\mathbf{h}\end{aligned},$$

where $\mathbf{h}=\sum_{j=1}^{m}\mathbf{h}_j\cdot\mathbf{f}/\mathbf{f}_j$.

The above fourth equality holds because $\mathbf{f}/\mathbf{f}_j\in R, j\in[\![m]\!]$ and $\|\mathbf{p}_0\|,\|\mathbf{h}_j\|,\|\mathbf{f}/\mathbf{f}_j\|$ all are small according to our parameter setting. That is, $\mathbf{u}=\mathbf{p}_0\cdot\mathbf{h}$ is not reduced modulo q. So, one can compute a basis $\mathbf{P}_0$ of $\mathbf{p}_0$ from $\mathbf{u},\mathbf{q}$ and a basis $\mathbf{F}$ of $\mathbf{f}$. However, the short generators for $\mathbf{p}_0,\mathbf{f}$ cannot at present be found using $\mathbf{P}_0$, $\mathbf{F}$ and $\mathbf{u},\mathbf{q}$.

For the averaging attacks considered in [GGH13, LS14], the current countermeasure is to increase dimension of ideal lattice of our scheme. The security of our scheme is based on the difficulty of finding any short element of the secret element $\mathbf{f}$.

3.3.2 Attack with Known f

If $\mathbf{f}$ is known, our construction is broken. Since $\mathbf{x}_i\bmod\mathbf{f}=\mathbf{d}_i$, $\mathbf{y}_i\bmod\mathbf{f}=\mathbf{c}_i$, we compute $\mathbf{g}=\mathbf{c}_1/\mathbf{d}_1\bmod\mathbf{f}$, where we assume that $\mathbf{d}_1$ is invertible over the modulo $\mathbf{f}$. Otherwise, we can use other $\mathbf{d}_i$'s. Then we have

$$\begin{aligned}\mathbf{g}\bmod\mathbf{f}_j&=(\mathbf{c}_1/\mathbf{d}_1)\bmod\mathbf{f}_j\\&=((\mathbf{c}_1\bmod\mathbf{f}_j)/(\mathbf{d}_1\bmod\mathbf{f}_j))\bmod\mathbf{f}_j\\&=(\mathbf{a}_{1,j}\mathbf{g}_j/\mathbf{a}_{1,j})\bmod\mathbf{f}_j\\&=\mathbf{g}_j\bmod\mathbf{f}_j\end{aligned}.$$

Given an arbitrary level- k encoding $\mathbf{u}$, we know $\mathbf{u}\bmod\mathbf{f}_j=\mathbf{a}_j(\mathbf{g}_j)^k\bmod\mathbf{f}_j, j\in[\![m]\!]$. So, $\mathbf{a}=(\mathbf{u}\bmod\mathbf{f})/\mathbf{g}^k$ is the level- 0 encoding of $\mathbf{u}$. This is because

$$\begin{aligned}\mathbf{a}\bmod\mathbf{f}_j&=(\mathbf{u}\bmod\mathbf{f})/\mathbf{g}^k\bmod\mathbf{f}_j\\&=(\mathbf{u}\bmod\mathbf{f}_j)/(\mathbf{g}^k\bmod\mathbf{f}_j)\bmod\mathbf{f}_j\\&=(\mathbf{a}_j(\mathbf{g}_j)^k)/((\mathbf{g}_j)^k)\bmod\mathbf{f}_j\\&=\mathbf{a}_j\bmod\mathbf{f}_j\end{aligned}.$$

Thus, $\mathbf{f}$ must be kept secret in our construction.

3.3.3 Attack for Small Multiple of f

When Small Multiple of $\mathbf{f}$ is known, one cannot attack our construction. Without loss of generality, assume $\mathbf{q}' = \mathbf{r} \cdot \mathbf{f}$ such that $\|\mathbf{r}\|$ is small. One can find a basis of $\mathbf{f}$ using $\mathbf{q}', \mathbf{q}$, but cannot efficiently find $\mathbf{f}$ applying current algorithms. Using $\mathbf{q}'$, one cannot also obtain the level-0 encoding of $\mathbf{u}$ by the method of known $\mathbf{f}$.

4 Variant of ideal multilinear maps

From the cryptanalysis above, there exist easily computable bases in our scheme. These bases are related to secret ring elements and can threaten our scheme security. This is because $\mathbf{q}$ is an encoding of zero in our scheme above. The reason why we must include this encoding of zero is to obtain ideal multilinear maps. Now, we describe a variant, whose method is to combine that in [CLT14, Gu15].

4.1 Construction

Setting the parameters. Because our construction uses the GGH construction as the basic component, our parameter setting is set as that of GGH to conveniently describe and compare. Let λ be the security parameter, n the dimension of elements of R. Concrete parameters are set as $\sigma = \sqrt{\lambda n}$, $\sigma' = \lambda n^{1.5}$, $q \ge 2^{\eta} n^{O(1)}$, $n = O(\lambda^2)$, $m = 2$, $l = O(\lambda^2)$, $\tau = O(n^2)$.

Instance generation: $(\mathrm{par}_2) \leftarrow \mathrm{InstGen}_2(1^{\lambda})$.

(1) Choose a large enough prime q.

(2) Sample $\mathbf{a}_{i,j}, \mathbf{f}_j, \mathbf{g}_j, \mathbf{h}_j \leftarrow D_{\mathbb{Z}^n,\sigma}$, $\mathbf{s}_i, \mathbf{t}_i \leftarrow D_{\mathbb{Z}^n,q}$, $i \in [\![\tau]\!]$, $j \in [\![m]\!]$ and $\mathbf{p}_0 \leftarrow D_{\mathbb{Z}^n,q}$ such that all ideal lattices $\mathbf{f}_j$'s are co-prime, $\mathbf{f}_j^{-1} \in \Bbbk$ and $\|\mathbf{f}_j^{-1}\| \le l$.

(3) Set $\mathbf{f} = \prod_{j=1}^{m} \mathbf{f}_j$ and $\mathbf{q} = \mathbf{p}_0 \cdot \mathbf{f}$.

(4) Compute $\mathbf{d}_i$, $\mathbf{c}_i$, $i \in [\![\tau]\!]$ over the modulo $\mathbf{f}$ such that $\mathbf{d}_i = \mathbf{a}_{i,j} \bmod \mathbf{f}_j$ and $\mathbf{c}_i = (\mathbf{a}_{i,j} \cdot \mathbf{g}_j) \bmod \mathbf{f}_j$, $j \in [\![m]\!]$.

(5) Set $\mathbf{x}_i = (\mathbf{d}_i + \mathbf{s}_i \cdot \mathbf{f}) \bmod \mathbf{q}$, and $\mathbf{y}_i = (\mathbf{c}_i + \mathbf{t}_i \cdot \mathbf{f}) \bmod \mathbf{q}$.

(6) Set $\mathbf{p}_{zt} = (\sum_{j=1}^{m} \mathbf{h}_j \cdot (\mathbf{f}_j^{-1} \bmod q)) \bmod q$.

(7) Sample $\mathbf{p}_i \leftarrow D_{\mathbb{Z}^n,q_i}$, $i \in [\![\mu - 1]\!]$, where $\mu = \lceil \log_{\sigma} q \rceil$ and $q_i = \sigma^{\mu - i}, i \in [\![\mu - 1]\!]$.

(8) Sample $\mathbf{e}_{i,j}, \mathbf{r}_{i,j} \leftarrow D_{\mathbb{Z}^n,\sigma}$, $i \in [\![\mu - 1]\!]$, $j \in [\![m]\!]$ and compute $\mathbf{e}_i$ such that $\mathbf{e}_i = \mathbf{e}_{i,j} \bmod \mathbf{f}_j$

(9) Set $\mathbf{q}_i = \mathbf{e}_i + \mathbf{p}_i \cdot \mathbf{f}$ and $\mathbf{p}_{zt,i} = (\sum_{j=1}^{m} \mathbf{h}_j (\mathbf{e}_{i,j} + \mathbf{r}_{i,j} \mathbf{f}_j)(\mathbf{f}_j^{-1} \bmod q)) \bmod q$, $i \in [\![\mu - 1]\!]$, such that $\|\mathbf{q}_i^{-1}\| \le n(\|\mathbf{q}_i\|)^{-1}$ and $\|\mathbf{q}_{i-1}\| \le n\sigma \|\mathbf{q}_i\|$. If $\|\mathbf{q}_i^{-1}\| > n(\|\mathbf{q}_i\|)^{-1}$, one resamples $\mathbf{p}_i \leftarrow D_{\mathbb{Z}^n,q_i}$.

(10) Output public parameters $\mathrm{par}_2 = \left\{q, \mathbf{q}, \left\{\mathbf{x}_i, \mathbf{y}_i\right\}_{i\in[\![\tau]\!]}, \mathbf{p}_{zt}, \left\{\mathbf{q}_i, \mathbf{p}_{zt,i}\right\}_{i\in[\![\mu-1]\!]}\right\}$.

Generating level-k random encodings: $\mathbf{u} \leftarrow \mathrm{Enc}_2(\mathrm{par}_2, k, \{\mathbf{w}_i\}_{i\in[\![\tau]\!]})$.

Choose $\mathbf{w}_i \leftarrow D_{\mathbb{Z}^n,\sigma'}$, $i \in [\![\tau]\!]$, generate a level-0 encoding $\mathbf{d} = \sum_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{x}_i)^k \bmod \mathbf{q}$ and a level-k encoding $\mathbf{u} = \sum_{i=1}^{\tau} \mathbf{w}_i \cdot (\mathbf{y}_i)^k \bmod \mathbf{q}$.

Adding encodings: $\mathbf{u} \leftarrow \mathrm{Add}_2(\mathrm{par}_2, k, \mathbf{u}_1, \cdots, \mathbf{u}_s)$.

Given s level-k encodings $\mathbf{u}_t, t \in [\![s]\!]$, their sum $\mathbf{u} = \sum_{t=1}^{s} \mathbf{u}_t \bmod \mathbf{q}$ is a level-k encoding.

Multiplying encodings: $\mathbf{u} \leftarrow \mathrm{Mul}_2(\mathrm{par}_2, 1, \mathbf{u}_1, \cdots, \mathbf{u}_k)$.

Given k level-1 encodings $\mathbf{u}_t$, $t \in [\![k]\!]$, their product $\mathbf{u} = \prod_{t=1}^{k} \mathbf{u}_t \bmod \mathbf{q}$ is a level-k encoding.

Zero Testing: $\mathrm{isZero}_2(\mathrm{par}_2, \mathbf{u}_0)$**.**

To determine whether level-k encoding $\mathbf{u}_0$ is an encoding of zero, compute as follows:

(1)For $i = 1$ to $\mu - 1$

Compute $\mathbf{u}_i = \mathbf{u}_{i-1} \bmod \mathbf{q}_i$ and $\mathbf{k}_i = (\mathbf{u}_{i-1} - \mathbf{u}_i) / \mathbf{q}_i$.

(2)Compute $\mathbf{v} = \left[\mathbf{p}_{zt} \cdot \mathbf{u}_{\mu-1} + \sum_{i=1}^{\mu-1} \mathbf{p}_{zt,i} \cdot \mathbf{k}_i\right]_q$

(3)Checked whether $\|\mathbf{v}\|$ is short:

$$\mathrm{isZero}_2(\mathrm{par}_2, \mathbf{u}_0) = \begin{cases} 1 & \text{if } \|\mathbf{v}\| < q / 2^{\eta} \\ 0 & \text{otherwise} \end{cases}.$$

Extract: $sk \leftarrow \mathrm{Ext}_2(\mathrm{par}_2, \mathbf{u}_0)$.

Given a level-k encoding $\mathbf{u}_0$, compute as follows:

(1) For $i = 1$ to $\mu - 1$

Compute $\mathbf{u}_i = \mathbf{u}_{i-1} \bmod \mathbf{q}_i$ and $\mathbf{k}_i = (\mathbf{u}_{i-1} - \mathbf{u}_i) / \mathbf{q}_i$.

(2)Compute $\mathbf{v} = \left[\mathbf{p}_{zt} \cdot \mathbf{u}_{\mu-1} + \sum_{i=1}^{\mu-1} \mathbf{p}_{zt,i} \cdot \mathbf{k}_i\right]_q$

(3)Extract the most significant bits $\mathrm{Ext}_2(\mathrm{par}_2, \mathbf{u}_0) = \mathrm{Extract}(\mathrm{msbs}_{\eta}(\mathbf{v}))$.

Remark 4.1 (1) If one sets $m = O(\lambda^2)$ and $n = 1$ for our variant, then the variant is extended to the ring $\mathbb{Z}$ of integers. In some sense this case is similar to the construction in [CLT15], however our construction is ideal multilinear maps, whereas their construction is approximate multilinear maps. Moreover, our variant needs to use the method of constructing new zero-testing parameter in [Gu15]. (2) One cannot set $m = O(\lambda^2)$ and $n = 1$ for the ideal multilinear maps in Section 3. This is because the multiplier $\mathbf{q}_0$ is an integer and can be computed in this case. (3) One can set $m = 1$ and $n = O(\lambda^2)$ for our variant to decrease time of computing inverse elements in generating instance algorithm.

4.2 Correctness

We first give Lemma 4.2 to show the correctness of our variant construction.

Lemma 4.2 If $\|\mathbf{p}^{-1}\| \le n(\|\mathbf{p}\|)^{-1}$ and $\|\mathbf{u}\| / \|\mathbf{p}\| \le \alpha$, then $\mathbf{u} \bmod \mathbf{p} = \mathbf{u} - \mathbf{k} \cdot \mathbf{p}$ and $\mathbf{k}$ is satisfied $\|\mathbf{k}\| \le n^2(\alpha+1)$.

Proof. Since $\mathbf{u} \bmod \mathbf{p} = \mathbf{u} - \mathbf{k}\cdot\mathbf{p}$, then $\mathbf{k} = \mathbf{p}^{-1}(\mathbf{u} - \mathbf{u} \bmod \mathbf{p})$. So , we have

$$\begin{aligned}\|\mathbf{k}\| &= \|\mathbf{p}^{-1}(\mathbf{u} - \mathbf{u} \bmod \mathbf{p})\| \\ &\le n\|\mathbf{p}^{-1}\|\|(\mathbf{u} - \mathbf{u} \bmod \mathbf{p})\| \\ &\le n\|\mathbf{p}^{-1}\|(\|\mathbf{u}\|_\infty + \|\mathbf{u} \bmod \mathbf{p}\|) \\ &\le n\|\mathbf{p}^{-1}\|(\|\mathbf{u}\| + \|\mathbf{p}\|) \\ &\le n\|\mathbf{p}^{-1}\|(\alpha\|\mathbf{p}\| + \|\mathbf{p}\|) \\ &\le n^2(\alpha+1)\end{aligned}.$$

So, $\|\mathbf{k}\|$ is a small integer. ■

Similar as that in the construction of ideal multilinear maps, it is easy to prove that $\mathrm{InstGen}_2$, Enc_2, Add_2, Mul_2 are correct. Here we only require to prove that Red_2, isZero_2 and Ext_2 are correct.

Lemma 4.3 For an arbitrary integer $k > 0$, the zero-testing algorithm $\mathrm{isZero}_2(\mathrm{par}_2, \mathbf{u}_0)$ correctly determines whether a level-k encoding $\mathbf{u}_0$ is an encoding of zero.

Proof. Given an arbitrary level-k encoding $\mathbf{u}_0$, we have $\mathbf{u}_0 = \mathbf{d} + \mathbf{r}\cdot\mathbf{f}$ and $\|\mathbf{u}_0\| < \|\mathbf{q}\|$ with $\|\mathbf{d}\| < \|\mathbf{f}\|$.

For $i \in [\![\mu-1]\!]$, $\mathbf{u}_i = \mathbf{u}_{i-1} \bmod \mathbf{q}_i$, then $\|\mathbf{u}_i\| \le \|\mathbf{q}_i\|$. So, we have

$$\|\mathbf{u}_{i-1}\| / \|\mathbf{q}_i\| \le \|\mathbf{q}_{i-1}\| / \|\mathbf{q}_i\| \le n\sigma.$$

By Lemma 4.2 and $\|\mathbf{q}_i^{-1}\| \le n(\|\mathbf{q}_i\|)^{-1}$, we have $\|\mathbf{k}_i\| \le n^2(n\sigma+1)$.

For $i \in [\![\mu-1]\!]$, $\mathbf{u}_i = \mathbf{u}_{i-1} \bmod \mathbf{q}_i$, $\mathbf{k}_i = (\mathbf{u}_{i-1} - \mathbf{u}_i)/\mathbf{q}_i$, then $\mathbf{u}_i = \mathbf{u}_{i-1} - \mathbf{k}_i\cdot\mathbf{q}_i$. So, we have

$$\mathbf{u}_0 = \mathbf{u}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{q}_i = \mathbf{u}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot(\mathbf{e}_i + \mathbf{p}_i\cdot\mathbf{f}).$$

Namely,

$$\mathbf{u}_0 \bmod \mathbf{f} = (\mathbf{u}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot(\mathbf{e}_i + \mathbf{p}_i\cdot\mathbf{f})) \bmod \mathbf{f} = (\mathbf{u}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_i) \bmod \mathbf{f},$$

$$\mathbf{u}_0 \bmod \mathbf{f}_j = (\mathbf{u}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_{i,j}) \bmod \mathbf{f}_j.$$

Moreover, we have

$$\begin{aligned}&\left\|\mathbf{u}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_i\right\| \\ &\le \|\mathbf{u}_{\mu-1}\| + \sum\nolimits_{i=1}^{\mu-1}\|\mathbf{k}_i\cdot\mathbf{e}_i\| \\ &\le \|\mathbf{u}_{\mu-1}\| + \sum\nolimits_{i=1}^{\mu-1}\|\mathbf{k}_i\|\|\mathbf{e}_i\| \\ &\le \|\mathbf{q}_{\mu-1}\| + \sum\nolimits_{i=1}^{\mu-1}\|\mathbf{k}_i\|\|\mathbf{f}\| \\ &\le (n\sigma + \mu n^3(n\sigma+1))\|\mathbf{f}\|\end{aligned},$$

$$\begin{aligned}&\left\|\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_{i,j}\right\|\\&\le\left\|\mathbf{u}_{\mu-1}\right\|+\sum\nolimits_{i=1}^{\mu-1}\left\|\mathbf{k}_i\cdot\mathbf{e}_{i,j}\right\|.\\&\le n\sigma\left\|\mathbf{f}\right\|+n\mu\left\|\mathbf{k}_i\right\|\left\|\mathbf{e}_{i,j}\right\|\\&\le n^4\sigma^3+n^3\sigma^2\mu\end{aligned}$$

Now, we have

$$\begin{aligned}\mathbf{v}&=\left[\mathbf{p}_{zt}\cdot\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{p}_{zt,i}\cdot\mathbf{k}_i\right]_q\\&=\left[(\sum\nolimits_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1}\bmod q)\cdot\mathbf{u}_{\mu-1})+\sum\nolimits_{i=1}^{\mu-1}(\sum\nolimits_{j=1}^{m}\mathbf{h}_j(\mathbf{e}_{i,j}+\mathbf{r}_{i,j}\mathbf{f}_j)(\mathbf{f}_j^{-1}\bmod q)\cdot\mathbf{k}_i)\right]_q.\\&=\left[\sum\nolimits_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1}\bmod q)\cdot(\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i(\mathbf{e}_{i,j}+\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q\end{aligned}$$

(1) If $\mathbf{u}_0$ is an encoding of zero, then $\mathbf{u}_0\bmod\mathbf{f}_j=0,\ j\in[\![m]\!]$, $\mathbf{u}\bmod\mathbf{f}=0$ and $\mathbf{d}=0$. So, we have $\mathbf{u}_0\bmod\mathbf{f}_j=(\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_{i,j})\bmod\mathbf{f}_j=0$. That is, $\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_{i,j}=\mathbf{t}_j\mathbf{f}_j$.

So, $\mathbf{v}=\left[\sum\nolimits_{j=1}^{m}\mathbf{h}_j(\mathbf{t}_j+\mathbf{r}_j)\right]_q=\sum\nolimits_{j=1}^{m}\mathbf{h}_j(\mathbf{t}_j+\mathbf{r}_j)$, where $\mathbf{r}_j=\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\mathbf{r}_{i,j}$.

Thus, we have

$$\begin{aligned}&\left\|\mathbf{v}\right\|\\&=\left\|\sum\nolimits_{j=1}^{m}\mathbf{h}_j(\mathbf{t}_j+\mathbf{r}_j)\right\|\\&\le m\left\|\mathbf{h}_j(\mathbf{t}_j+\mathbf{r}_j)\right\|\\&\le mn\left\|\mathbf{h}_j\right\|(\left\|\mathbf{t}_j\right\|+\left\|\mathbf{r}_j\right\|)\qquad.\\&\le 2n^2\sigma\left\|(n^4\sigma^3+n^3\sigma^2\mu+\mu n^3\sigma^2)\right\|\\&\le n^{O(1)}\\&\le q/2^{\eta}\end{aligned}$$

(2) If $\mathbf{u}_0$ is not an encoding of zero, then $\mathbf{u}_0\bmod\mathbf{f}\neq0$. That is, $\exists j\in[\![m]\!],\mathbf{d}\bmod\mathbf{f}_j\neq0$. So, $\mathbf{u}_0\bmod\mathbf{f}_j=(\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_{i,j})\bmod\mathbf{f}_j=\mathbf{d}\bmod\mathbf{f}_j$, namely $\mathbf{u}_{\mu-1}+\sum\nolimits_{i=1}^{\mu-1}\mathbf{k}_i\cdot\mathbf{e}_{i,j}=\mathbf{t}_j\mathbf{f}_j+\mathbf{d}$.

Thus, we have

$$\begin{aligned}\|\mathbf{v}\| &= \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1} \bmod q)\cdot(\mathbf{u}_{\mu-1}+\sum_{i=1}^{\mu-1}\mathbf{k}_i(\mathbf{e}_{i,j}+\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q\right\| \\ &= \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1} \bmod q)\cdot(\mathbf{u}_{\mu-1}+\sum_{i=1}^{\mu-1}\mathbf{k}_i\mathbf{e}_{i,j}+\sum_{i=1}^{\mu-1}\mathbf{k}_i\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q\right\| \\ &= \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1} \bmod q)\cdot(\mathbf{d}+\mathbf{t}_j\mathbf{f}_j+\sum_{i=1}^{\mu-1}\mathbf{k}_i\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q\right\| \\ &= \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j\mathbf{d}(\mathbf{f}_j^{-1} \bmod q)+\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{t}_j+\mathbf{r}_j)\right]_q\right\| \qquad , \\ &\geq \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j\mathbf{d}(\mathbf{f}_j^{-1} \bmod q)\right]_q\right\| - \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{t}_j+\mathbf{r}_j)\right]_q\right\| \\ &\geq q^{1-\varepsilon} - q/2^{\eta} \\ &\geq q^{1-\varepsilon'}\end{aligned}$$

where $\mathbf{r}_j = \sum_{i=1}^{\mu-1}\mathbf{k}_i\mathbf{r}_{i,j}$.

So, $\mathrm{isZero}_2(\mathrm{par}_2,\mathbf{u}_0)$ can correctly decide whether the encoding of $\mathbf{u}_0$ is zero. ■

Lemma 4.4 If two level-k encodings $\mathbf{u}_0^{(1)},\mathbf{u}_0^{(2)}$ encode same level-0 element, then $\mathrm{Ext}_2(\mathrm{par}_2,\mathbf{u}_0^{(1)}) = \mathrm{Ext}_2(\mathrm{par}_2,\mathbf{u}_0^{(2)})$.

Proof. Since $\mathbf{u}_0^{(1)} = \mathbf{u}_0^{(2)} = \mathbf{a}_j(\mathbf{g}_j)^k \bmod \mathbf{f}_j, \forall j \in [\![m]\!]$ and $\mathbf{f}_j, j \in [\![m]\!]$ are co-prime, we get $\mathbf{u}_0^{(1)} = \mathbf{d}+\mathbf{r}^{(1)}\cdot\mathbf{f}, \mathbf{u}_0^{(2)} = \mathbf{d}+\mathbf{r}^{(2)}\cdot\mathbf{f}$.

For $i \in [\![\mu-1]\!]$, $\mathbf{u}_i^{(1)} = \mathbf{u}_{i-1}^{(1)} \bmod \mathbf{q}_i$, $\mathbf{k}_i^{(1)} = (\mathbf{u}_{i-1}^{(1)} - \mathbf{u}_i^{(1)})/\mathbf{q}_i$, we have $\mathbf{u}_i^{(1)} = \mathbf{u}_{i-1}^{(1)} - \mathbf{k}_i^{(1)}\cdot\mathbf{q}_i$.

So,

$$\mathbf{u}_0^{(1)} = \mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\cdot\mathbf{q}_i = \mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\cdot(\mathbf{e}_i+\mathbf{p}_i\cdot\mathbf{f}) .$$

$$\begin{aligned}&\mathbf{u}_0^{(1)} \bmod \mathbf{f} \\ &= (\mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\cdot(\mathbf{e}_i+\mathbf{p}_i\cdot\mathbf{f})) \bmod \mathbf{f} \qquad , \\ &= \mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\cdot\mathbf{e}_i) \bmod \mathbf{f} \\ &= \mathbf{d} \bmod \mathbf{f}\end{aligned}$$

$$\mathbf{u}_0^{(1)} \bmod \mathbf{f}_j = (\mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\cdot\mathbf{e}_{i,j}) \bmod \mathbf{f}_j = \mathbf{d} \bmod \mathbf{f}_j .$$

Namely, we get $\mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\cdot\mathbf{e}_{i,j} = \mathbf{d}+\mathbf{t}_i^{(1)}\mathbf{f}$.

So, we have

$$\begin{aligned}\mathbf{v}^{(1)} &= \left[\mathbf{p}_{zt}\cdot\mathbf{u}_{\mu-1}^{(1)} + \sum_{i=1}^{\mu-1}\mathbf{p}_{zt,i}\cdot\mathbf{k}_i^{(1)}\right]_q \\ &= \left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1} \bmod q)\cdot(\mathbf{u}_{\mu-1}^{(1)}+\sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}(\mathbf{e}_{i,j}+\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q \\ &= \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1} \bmod q)\cdot(\mathbf{u}_{\mu-1}^{(1)}+\sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\mathbf{e}_{i,j}+\sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q\right\| . \\ &= \left\|\left[\sum_{j=1}^{m}\mathbf{h}_j(\mathbf{f}_j^{-1} \bmod q)\cdot(\mathbf{d}+\mathbf{t}_j^{(1)}\mathbf{f}_j+\sum_{i=1}^{\mu-1}\mathbf{k}_i^{(1)}\mathbf{r}_{i,j}\mathbf{f}_j)\right]_q\right\|\end{aligned}$$

Similarly, we have

$$\mathbf{v}^{(2)} = \left[\mathbf{p}_{zt} \cdot \mathbf{u}^{(2)}_{\mu-1} + \sum\nolimits_{i=1}^{\mu-1} \mathbf{p}_{zt,i} \cdot \mathbf{k}_i^{(2)}\right]_q$$

$$= \left\| \left[\sum\nolimits_{j=1}^{m} \mathbf{h}_j (\mathbf{f}_j^{-1} \bmod q) \cdot (\mathbf{d} + \mathbf{t}_j^{(2)} \mathbf{f}_j + \sum\nolimits_{i=1}^{\mu-1} \mathbf{k}_i^{(2)} \mathbf{r}_{i,j} \mathbf{f}_j)\right]_q \right\| .$$

By Lemma 4.3, we have

$$\sum\nolimits_{j=1}^{m} \mathbf{h}_j (\mathbf{t}_j^{(1)} + \sum\nolimits_{i=1}^{\mu-1} \mathbf{k}_i^{(1)} \mathbf{r}_{i,j}) \le q / 2^{\eta} ,$$

$$\sum\nolimits_{j=1}^{m} \mathbf{h}_j (\mathbf{t}_j^{(2)} + \sum\nolimits_{i=1}^{\mu-1} \mathbf{k}_i^{(2)} \mathbf{r}_{i,j}) \le q / 2^{\eta} .$$

Hence, the most significant bits of $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$ are all decided by the second term $(\sum\nolimits_{j=1}^{m} (\mathbf{h}_j \cdot \mathbf{d}) \cdot (\mathbf{f}_j^{-1} \bmod q))$. Namely, $\mathrm{Ext}_1(\mathrm{par}_1, \mathbf{u}_1) = \mathrm{Ext}_1(\mathrm{par}_1, \mathbf{u}_2)$. ■

4.3 Security

Variant security depends on new assumptions, and the extraction variant of GCDH/GDDH is called ext-GCDH/ext-GDDH. The ext-GCDH/ext-GDDH is introduced in [LSS14] to prove the security of the GGHLite scheme. We describe the security experiment of our variant below:

(1) $(\mathrm{par}_2) \leftarrow \mathrm{InstGen}_2(1^{\lambda})$.

(2) Choose an arbitrary positive integer k .

(3) For $t = 0$ to k :

Sample $\mathbf{w}_{t,i} \leftarrow D_{\mathbb{Z}^n,\sigma'}, i \in [\![\tau]\!]$

Generate level-1 encoding of $\mathbf{d}_t = (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_{t,i} \cdot \mathbf{x}_i) \bmod \mathbf{q}$:

$\mathbf{u}_t = (\sum\nolimits_{i=1}^{\tau} \mathbf{w}_{t,i} \cdot \mathbf{y}_i) \bmod \mathbf{q}$.

(4) Sample $\mathbf{r}_{0,i} \leftarrow D_{\mathbb{Z}^n,\sigma'}, i \in [\![\tau]\!]$ and generate $\mathbf{r}_0 = (\sum\nolimits_{i=1}^{\tau} \mathbf{r}_{0,i} \cdot \mathbf{x}_i) \bmod \mathbf{q}$.

(5) Compute $\mathbf{u}^* = \prod\nolimits_{t=1}^{k} \mathbf{u}_t \bmod \mathbf{q}$, $\mathbf{u} = (\mathbf{d}_0 \cdot \mathbf{u}^*) \bmod \mathbf{q}$ and $\mathbf{u}' = \mathbf{r}_0 \cdot \mathbf{u}^* \bmod \mathbf{q}$.

(6) Set $\mathbf{v}_C = \mathbf{v}_D = \mathrm{Ext}_2(\mathrm{par}_2, \mathbf{u})$.

(7) Set $\mathbf{v}_R = \mathrm{Ext}_2(\mathrm{par}_2, \mathbf{u}')$.

Definition 4.5 (ext-GCDH/ext-GDDH). The extraction k-graded CDH problem (ext-GCDH) is, on input $\{\mathrm{par}_2, \mathbf{u}_0, \cdots, \mathbf{u}_k\}$, to output an extraction encoding $\mathbf{w} \in R_p$, such that $\mathrm{Ext}_2(\mathrm{par}_2, \mathbf{w}) = \mathbf{v}_C$. The extraction k-graded DDH problem (ext-GDDH) distinguishes between $\mathbf{v}_D$ and $\mathbf{v}_R$, that is, between the distributions $D_{GDDH} = \{\mathrm{par}_2, \mathbf{u}_0, \cdots, \mathbf{u}_k, \mathbf{v}_D\}$ and $D_{RAND} = \{\mathrm{par}_2, \mathbf{u}_0, \cdots, \mathbf{u}_k, \mathbf{v}_R\}$.

5 Asymmetric ideal multilinear maps

Asymmetric ideal multilinear maps with different group are required in some applications. Similar to [GGH13], we briefly describe asymmetric variant as follow.

In this variant, we sample different $\mathbf{g}_{j,k} \leftarrow D_{\mathbb{Z}^n,\sigma}, k = 1, ..., \beta$. The element of the form $(\mathbf{a}_{j,k} \cdot \mathbf{g}_{j,k}) \bmod \mathbf{f}_j$ is a level-1 encoding relative to the k-th generator $\mathbf{g}_{j,k}$. We denote by vectors the different levels of encoding. For a level-0 encoding $\mathbf{a}$, the encoding $\mathbf{c}$ with an index vector $\mathbf{w} = (w_1, ..., w_{\beta}) \in \mathbb{N}^{\beta}$ is satisfied to

$\mathbf{c} \bmod \mathbf{f}_j = (\mathbf{a} \prod_{k=1}^{\beta} (\mathbf{g}_{j,k})^{w_k}) \bmod \mathbf{f}_j$. So, we give the public parameters $\mathbf{x}_{i,k} = (\mathbf{d}_{i,k} + \mathbf{s}_{i,k} \cdot \mathbf{f}) \bmod \mathbf{q}$ and $\mathbf{y}_{i,k} = (\mathbf{c}_{i,k} + \mathbf{t}_{i,k} \cdot \mathbf{f}) \bmod \mathbf{q}$, $i \in [\![\tau]\!]$, $k \in [\![\beta]\!]$, where $\mathbf{d}_{i,k} = \mathbf{a}_{i,j,k} \bmod \mathbf{f}_j$, $\mathbf{c}_{i,k} = (\mathbf{a}_{i,j,k} \cdot \mathbf{g}_{j,k}) \bmod \mathbf{f}_j$. That is, the public parameters in asymmetric variant is $\mathrm{par}_1 = \left\{ q, \mathbf{q}, \left\{ \mathbf{x}_{i,k}, \mathbf{y}_{i,k} \right\}_{i \in [\![\tau]\!], k \in [\![\beta]\!]}, \mathbf{p}_{zt} \right\}$. For the variant in Section 4, we can similarly obtain asymmetric variant, namely $\mathrm{par}_2 = \left\{ q, \mathbf{q}, \left\{ \mathbf{x}_{i,k}, \mathbf{y}_{i,k} \right\}_{i \in [\![\tau]\!], k \in [\![\beta]\!]}, \mathbf{p}_{zt}, \left\{ \mathbf{q}_i, \mathbf{p}_{zt,i} \right\}_{i \in [\![\mu-1]\!]} \right\}$.

6 One-round Multipartite Diffie-Hellman Key Exchange Protocol

We describe the construction of one-round multipartite Diffie-Hellman key exchange protocol using our ideal multilinear maps. As in [GGH13], protocol security relies on the hardness assumption of the ext-GDDH.

$\mathbf{Setup}(1^\lambda)$. For $\forall t \in [\![2]\!]$, output $(\mathrm{par}_t) \leftarrow \mathrm{InstGen}_t(1^\lambda)$ as the public parameter.

$\mathbf{Publish}(\mathrm{par}_t, j)$. Let N be the number of participants, $j \in [\![N]\!]$. Each party j samples random elements $\mathbf{w}_{j,i} \leftarrow D_{\mathbb{Z}^n, \sigma'}$, $i \in [\![\tau]\!]$, computes level- 0 encoding $\mathbf{d}_j = \sum_{i=1}^{\tau} \mathbf{w}_{j,i} \cdot (\mathbf{x}_i)^k \bmod \mathbf{q}$ as a secret key, and publishes level-1 encoding $\mathbf{u}_j \leftarrow \mathrm{Enc}_t \left(\mathrm{par}_t, 1, \left\{ \mathbf{w}_{j,i} \right\}_{i \in [\![\tau]\!]} \right)$ as a public key.

$\mathbf{KeyGen}(\mathrm{par}_t, j, \mathbf{d}_j, \left\{ \mathbf{u}_j \right\}_{j \neq i})$. Each party j computes $\mathbf{c}_j = \mathbf{d}_j \times \prod_{k \neq j} \mathbf{u}_k$ and extracts the common secret key $sk = \mathrm{Ext}_t(\mathrm{par}_t, \mathbf{c}_j)$.

Theorem 6.1 Suppose that ext-GDDH is hard, then our construction above is one-round multipartite Diffie-Hellman key exchange protocol.

Proof. The proof is similar as Theorem 2 in [GGH13]. ■

References

[BF03] D. Boneh and M. K. Franklin. Identity-based encryption from the Weil pairing, SIAM Journal on Computing, 32(3):586–615, 2003.

[BS03] D. Boneh and A. Silverberg. Applications of multilinear forms to cryptography. Contemporary Mathematics, 324:71–90, 2003.

[BWZ14] D. Boneh, D. J. Wu, and J. Zimmerman. Immunizing multilinear maps against zeroizing attacks. http://eprint.iacr.org/2014/930.

[CHL+14] J. H. Cheon, K. Han, C. Lee, H. Ryu, D. Stehle. Cryptanalysis of the Multilinear Map over the Integers. http://eprint.iacr.org/2014/906.

[CN11] Y. Chen and P. Q. Nguyen. BKZ 2.0 Better Lattice Security Estimates, ASIACRYPT 2011, LNCS 7073, pp. 1–20.

[CLT13] J. S. Coron, T. Lepoint, and M. Tibouchi. Practical multilinear maps over the integers. CRYPTO 2013, LNCS 8042, pp. 476–493.

[CLT14] J. S. Coron, T. Lepoint, and M. Tibouchi. Cryptanalysis of two candidate fixes of multilinear maps over the integers. http://eprint.iacr.org/2014/975.

[CLT15] J. S. Coron, T. Lepoint, and M. Tibouchi. New Multilinear Maps over the Integers. http://eprint.iacr.org/2015/162.

[GGH13] S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. EUROCRYPT 2013, LNCS 7881, pp. 1–17.

[GGHZ14] S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Fully secure functional encryption without obfuscation. http://eprint.iacr.org/2014/666.

[GLSW14] C. Gentry, A. Lewko, A. Sahai, and B. Waters. Indistinguishability obfuscation from the multilinear subgroup elimination assumption. http://eprint.iacr.org/2014/309.

[Gu15] Gu Chunsheng. Multilinear Maps Using Ideal Lattices without Encodings of Zero. http://eprint.iacr.org/2015/023.

[HPS98] J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: a ring based public key cryptosystem. ANTS 1998, LNCS 1423, pp. 267-288.

[Jou00] A. Joux. A one round protocol for tripartite Diffie-Hellman. ANTS 2000, LNCS 1838, pp. 385–394.

[LSS14] A. Langlois, D. Stehlé, and R. Steinfeld, GGHLite: More Efficient Multilinear Maps from Ideal Lattices, EUROCRYPT 2014, LNCS 8441, 2014, pp. 239–256.

[PTT10] C. Papamanthou, R. Tamassia, and N. Triandopoulos. Optimal authenticated data structures with multilinear forms. Pairing 2010, LNCS 6487, pp. 246–264.

[Rot13] R. Rothblum. On the circular security of bit-encryption. TCC 2013, LNCS 7785, 2013, pp. 579–598.

[RS09] M. Rückert and D. Schröder. Aggregate and verifiably encrypted signatures from multilinear maps without random oracles. ISA 2009, LNCS 5576, pp. 750–759.

[Sma03] Smart, N.P. An identity based authenticated key agreement protocol based on the Weil pairing, Electronics Letters, 38(13), pp. 630-632, 2002.

[SOK00] R. Sakai, K. Ohgishi and M. Kasahara. Cryptosystems based on pairing, the 2000 Symposium on Cryptography and Information Security, Okinawa, Japan, 2000.

[SS11] D. Stehlé and R. Steinfeld. Making NTRU as secure as worst-case problems over ideal lattices, EUROCRYPT 2011, LNCS 6632, pp. 27–47.