Variation of GGH15 Multilinear Maps

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

Jan 1, 2015

Abstract. Recently, Coron presented an attack of GGH15 multilinear maps, which breaks the multipartite Diffie-Hellman key exchange protocol based on GGH15. In this paper, we describe a variation of GGH15, which seems to thwart known attacks.
Keywords. Multilinear maps, zeroizing attack, MPKE, witness encryption, LWE

1 Introduction

Multilinear maps have many applications including one-round multipartite key exchange [GGH13, BZ14], witness encryption [GGSW13] and program obfuscation [GGH+13a]. The notion of cryptographic multilinear maps was introduced by Boneh and Silverberg in 2003 [BS03]. However until 2013, the first candidate construction of multilinear maps was described over ideal lattices Garg, Gentry and Halevi [GGH13] (GGH13, for short). Then, the construction over the integers was presented by Coron, Lepoint and Tibouchi [CLT13] (CLT13, for short). Recently, the construction from lattices is proposed by Gentry, Gorbunov and Halevi [GGH15]. However, current constructions of multilinear maps [GGH13, CLT13, GGH15] suffer from zeroizing attacks [GGH13, CHL+15, CGH+15, HJ15, CLR15, GGH15, Cor15].

Attacks of CLT13. Cheon, Han, Lee, Ryu and Stehle [CHL+15] presented an extension of zeroizing attack, which completely breaks CLT13. To immune zeroizing attack, two fixes of CLT13 are proposed by Garg, Gentry, Halevi and Zhandry [GGH+14], and Boneh, Wu and Zimmerman [BWZ14], respectively. However, the fixes [GGH+14, BWZ14] were shown to be insecure in [CGH+15] by using an extension of Cheon et al.'s attack. By designing new zero-testing parameter, a new variant of CLT13 was described by Coron, Lepoint and Tibouchi [CLT15] (CLT15 for short). Unfortunately, CLT15 was recently broken independently by Cheon, Lee and Ryu [CLR15], and Minaud and Fouque [MF15].

Attacks of GGH13. Hu and Jia [HJ15a] described an efficient weak-DL-based attack on the GGH13, which breaks multipartite key-exchange protocol based on GGH13. A fix of GGH13 was recently proposed by Gentry, Halevi and Lepoint [Hal15] by replacing the linear zero-testing procedure from GGH13 with a quadratic (or higher-degree) procedure. But this new variant of GGH13 failed to immune zeroizing attack [BGH+15].

Attack of GGH15. Very recently, Coron [Cor15] described an attack which breaks the multipartite Diffie-Hellman key exchange protocol based on GGH15 multilinear maps.

An open problem is to fix previous constructions to obtain a secure multilinear map. In this paper, we presented a variation of GGH15, which seems to thwart known attacks.

1.1 Our variation

We describe a new multilinear map from lattices. Our variation follows same technique of GGH15 multilinear maps, but does not use directed acyclic graph. The GGH15 construction is parameterized by a directed acyclic graph $G=(V,E)$. Each node $v\in V$ is assigned with a random matrix $\mathbf{A}_v\in\mathbb{Z}_q^{m\times n}$. The edge $u\to v$ in E is assigned with an encoding $\mathbf{D}\in\mathbb{Z}_q^{m\times m}$ such that $\mathbf{D}\cdot\mathbf{A}_u=\mathbf{A}_v\cdot\mathbf{S}+\mathbf{E}$, where $\mathbf{S}\in\mathbb{Z}_q^{n\times n}$ is a secret plaintext matrix, $\mathbf{E}\leftarrow D_{\mathbb{Z}^{m\times n},\sigma}$ a small noise matrix.

The start point of our work is to replace the above secret plaintext matrix with special matrix generating by some ring element. As a result, our variation does not require a directed graph. However, since the secret plaintext matrices in GGH15 do not support commutative law, the GGH15 construction introduces a directed acyclic graph to obtain multipartite key exchange protocol.

Our variation is different from the GGH15 construction in the three points. (1) We modify the

general matrix $\mathbf{S}$ into an anti-cycle matrix generating by some ring element $\mathbf{s} \in R = \mathbb{Z}[x]/(x^n+1)$. (2) We set $\mathbf{A}_u = \mathbf{A}_v$ and use $\mathbf{D} \cdot \mathbf{A} = \mathbf{A} \cdot Rot(\mathbf{s}) + \mathbf{E}$. As a result, we do not need the directed acyclic graph using by the GGH15 construction. (3) To obtain graded encoding, our variation is to introduce a ring element $\mathbf{g} \in R$ and set $\mathbf{S} = Rot(\mathbf{g} \cdot \mathbf{s})$, whereas GGH15 uses the length of the longest directed path in the graph G. As a result, our variation seems to thwart current known attacks for GGH15.

Organization. Section 2 recalls some background. Section 3 describes our variation of GGH15. Section 4 gives security analysis for our variation. Finally, Section 5 presents multipartite key exchange (MPKE) and witness encryption (WE) based on our construction.

2 Preliminaries

2.1 Notations

We denote $\mathbb{Z}, \mathbb{Q}, \mathbb{R}$ the ring of integers, the field of rational numbers, and the field of real numbers. We take n as a positive integer and a power of 2. Notation $[\![n]\!]$ denotes the set $\{1, 2, \cdots, n\}$, and $[a]_q \in (-q/2, q/2]$. Vectors and matrices are denoted in bold, such as $\mathbf{a}, \mathbf{b}, \mathbf{c}$ and $\mathbf{A}, \mathbf{B}, \mathbf{C}$. The i-th entry of $\mathbf{a}$ is denoted as a_i, the element of the i-th row and j-th colomn of $\mathbf{A}$ is denoted as $a_{i,j}$. Similarly, $[\mathbf{a}]_q$, $[\mathbf{A}]_q$ denote $[a_i]_q$, $[a_{i,j}]_q$, respectively. Notation $\|\mathbf{a}\|_\infty$ ($\|\mathbf{a}\|$ for short) denotes the maximum norm of $\mathbf{a}$.

2.2 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 $\mathbf{B} \in \mathbb{R}^{n \times n}$, then $L = \{\mathbf{B}\mathbf{y} : \mathbf{y} \in \mathbb{Z}^n\}$. We say that $\mathbf{B}$ spans L if $\mathbf{B}$ is a basis for L. Given a basis $\mathbf{B}$ of L, we define $P(\mathbf{B}) = \{\mathbf{B}\mathbf{y} \mid \mathbf{y} \in \mathbb{R}^n, \forall i : -1/2 \le y_i < 1/2\}$ as the parallelization corresponding to $\mathbf{B}$. Let $\det(\mathbf{B})$ be the determinant of $\mathbf{B}$.

Given $\mathbf{c} \in \mathbb{R}^n$, $\sigma > 0$, the Gaussian distribution of a lattice L is defined as $\forall \mathbf{x} \in L$, $D_{L,\sigma,\mathbf{c}} = \rho_{\sigma,\mathbf{c}}(\mathbf{x}) / \rho_{\sigma,\mathbf{c}}(L)$, where $\rho_{\sigma,\mathbf{c}}(\mathbf{x}) = \exp(-\pi \|\mathbf{x} - \mathbf{c}\|^2 / \sigma^2)$, $\rho_{\sigma,\mathbf{c}}(L) = \sum_{x \in L} \rho_{\sigma,\mathbf{c}}(\mathbf{x})$. For simplicity, we write $D_{\mathbb{Z}^n,\sigma,0}$ as $D_{\mathbb{Z}^n,\sigma}$. Let $\mathbf{x} \leftarrow D_{L,\sigma}$ be a Gaussian sample over the lattice.

2.3 Multilinear Maps

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

(1) Elements $\{g_j \in G_j\}_{j=1,...,d}$, index $j \in [\![d]\!]$, and integer $a \in \mathbb{Z}_q$ hold that

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

(2) Map e is non-degenerate in the following sense: if elements $\{g_j \in G_j\}_{j=1,...,d}$ are

generators of their respective groups, then $e(g_1,\cdots,g_d)$ is a generator of G_T.

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

(1) For every index $j\in[\![d]\!]$, 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 $j\in[\![d]\!]$, 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 '$\times$' exists, such that every α_1,α_2, every index $j_1,j_2\in[\![d]\!]$ with $j_1+j_2\le\kappa$, 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 Our construction

Setting the parameters. For simplicity, our parameters setting is adaptively set to that in GGH15. Let λ be the security parameter, d the multilinearity level, $n=\Theta(d\lambda\log(d\lambda))$, $q=(d\lambda)^{\Theta(d)}$, $m=\Theta(nd\log q)$, $N=m^2+\Theta(\lambda)$, $s=\sqrt{n}$, $\sigma=\sqrt{n(d+1)\log q}$ and $v=\lfloor(\log q)/4\rfloor-1$.

3.1 Construction

Instance generation: $(\text{par})\leftarrow\text{InstGen}(1^\lambda,1^d)$.

(1) Choose a prime $q=(d\lambda)^{\Theta(d)}$.

(2) Sample $\mathbf{g}\leftarrow D_{\mathbb{Z}^n,\sigma}$, $\mathbf{v}\leftarrow D_{\mathbb{Z}^n,\sigma}$.

(3) Generate matrix $\mathbf{A}\in\mathbb{Z}_q^{m\times n}$ and any small enough full rank trapdoor matrix $\mathbf{T}\in\mathbb{Z}^{m\times m}$ such that $\left[\mathbf{T}\cdot\mathbf{A}\right]_q=0$ using TrapSamp in Lemma 2.1 [GGH15].

(4) Sample $\mathbf{s}_i\leftarrow D_{\mathbb{Z}^n,\sigma}$, $\mathbf{E}_i,\mathbf{F}_i\leftarrow D_{\mathbb{Z}^{m\times n},\sqrt{q}}$

(5) Set $\mathbf{V}_i=\left[\mathbf{A}\cdot Rot(\mathbf{gs}_i)+\mathbf{F}_i\right]_q$,

Sample $\mathbf{D}_i\leftarrow\text{PreSample}(\mathbf{A},\mathbf{T},\mathbf{V}_i,\sigma)$ such that $\left[\mathbf{D}_i\cdot\mathbf{A}\right]_q=\mathbf{V}_i$ using PreSample in Lemma 2.1 [GGH15].

Set $\mathbf{C}_i=\left[\mathbf{A}\cdot Rot(\mathbf{s}_i)+\mathbf{E}_i\right]_q$, and $\mathbf{c}_i=\left[\mathbf{C}_i\cdot\mathbf{v}\right]_q$.

(6) Output the public parameters $\text{par}=\left\{q,\left\{\mathbf{D}_i,\mathbf{c}_i\right\}_{i\in[\![N]\!]}\right\}$.

Generating level-1 encoding: $\mathbf{U}\leftarrow\text{Enc}\left(\text{par},\mathbf{r}\leftarrow D_{\mathbb{Z}^N,\sigma}\right)$.

Sample a vector $\mathbf{r}\leftarrow D_{\mathbb{Z}^N,\sigma}$, and generate a level-$1$ encoding $\mathbf{U}=\sum_{i=1}^N r_i\cdot\mathbf{D}_i$.

Adding encodings: $\mathbf{U}\leftarrow\text{Add}\left(\text{par},k,\mathbf{U}_1\cdots,\mathbf{U}_t\right)$.

Given level-k encodings $\mathbf{U}_l,l\in[\![t]\!]$, compute a level-$k$ encoding $\mathbf{U}=\sum_{l=1}^t\mathbf{U}_l$.

Multiplying encodings: $\mathbf{U}\leftarrow\text{Mul}\left(\text{par},1,\mathbf{U}_1,\cdots,\mathbf{U}_k\right)$.

Given level-1 encodings $\mathbf{U}_l, l \in [\![k]\!]$, compute a level-$k$ encoding $\mathbf{U} = \prod_{l=1}^{k} \mathbf{U}_l$.

Zero testing: $\mathrm{isZero}(\mathrm{par}, \mathbf{U}, \mathbf{r})$.

(1) Given $\mathbf{r} \leftarrow D_{\mathbb{Z}^N,\sigma}$, we compute $\mathbf{c} = \left[\sum_{i=1}^{N} r_i \cdot \mathbf{c}_i\right]_q$.

(2) Given a level-d encoding $\mathbf{U}$, we compute $\mathbf{w} = [\mathbf{U} \cdot \mathbf{c}]_q$ and check whether $\|\mathbf{w}\|$ is short:

$$\mathrm{isZero}(\mathrm{par}, \mathbf{U}) = \begin{cases} 1 & \text{if } \|\mathbf{w}\| < q^{3/4} \\ 0 & \text{otherwise} \end{cases}.$$

Extraction: $sk \leftarrow \mathrm{Ext}(\mathrm{par}, \mathbf{U}, \mathbf{r})$.

(1) Given $\mathbf{r} \leftarrow D_{\mathbb{Z}^N,\sigma}$, we compute $\mathbf{c} = \left[\sum_{i=1}^{N} r_i \cdot \mathbf{c}_i\right]_q$.

(2) Given a level-d encoding $\mathbf{U}$, we compute $\mathbf{w} = [\mathbf{U} \cdot \mathbf{c}]_q$ and collect v most-significant bits of each entry of $\mathbf{w}$:

$$\mathrm{Ext}(\mathrm{par}, \mathbf{U}, \mathbf{r}) = \mathrm{Extract}_s\left(\mathrm{msbs}_v(\mathbf{w})\right).$$

Remark 3.1 (1) The level of graded encodings in this variation is hidden, and is corresponding to the number of multiplying by encodings. Moreover, the variation can decide if a graded encoding with level less than or equal d encodes zero. (2) Using the safeguards in [GGH15], one can choose uniformly a invertible matrix $\mathbf{B} \in \mathbb{Z}_q^{m \times m}$ and a small vector $\mathbf{u} \leftarrow D_{\mathbb{Z}^m,\sigma}$, modify $\mathbf{D}_i, \mathbf{c}_i$ into $\mathbf{D}_i' = \left[\mathbf{B} \cdot \mathbf{D}_i \cdot \mathbf{B}^{-1}\right]_q$, $\mathbf{c}_i' = [\mathbf{B} \cdot \mathbf{c}_i]_q$, and add a new vector $\mathbf{u}' = [\mathbf{u} \cdot \mathbf{B}]_q$. In this case, the public parameters become $\mathrm{par} = \left\{q, \mathbf{u}', \left\{\mathbf{D}_i', \mathbf{c}_i'\right\}_{i \in [\![N]\!]}\right\}$. It is easy to verify that its correctness directly follows from the correctness of our variation.

3.2 Correctness

Lemma 3.2 The algorithm $\mathrm{InstGen}(1^\lambda, 1^d)$ runs in polynomial time.

Proof. By Lemma 2.1 [GGH15], TrapSamp and PreSample are polynomial time algorithms. All other steps also require polynomial time. □

Lemma 3.3 The encoding $\mathbf{U} \leftarrow \mathrm{Enc}\left(\mathrm{par}, \mathbf{r} \leftarrow D_{\mathbb{Z}^N,\sigma}\right)$ is a level-1 encoding.

Proof. Without loss of generality, let $\mathbf{c} = \left[(\mathbf{A} \cdot Rot(\mathbf{s}) + \mathbf{E}) \cdot \mathbf{v}\right]_q$.

By $\mathbf{U} = \sum_{i=1}^{N} r_i \cdot \mathbf{D}_i$ and $\mathbf{D}_i \cdot \mathbf{A} = \left(\mathbf{A} \cdot Rot(\mathbf{g}\mathbf{s}_i) + \mathbf{F}_i\right) \bmod q$, we have

$$\begin{aligned} [\mathbf{U} \cdot \mathbf{c}]_q &= \left[\sum_{i=1}^{N} r_i \cdot \mathbf{D}_i \cdot \mathbf{c}\right]_q \\ &= \left[\sum_{i=1}^{N} r_i \cdot \mathbf{D}_i \cdot (\mathbf{A} \cdot Rot(\mathbf{s}) + \mathbf{E}) \cdot \mathbf{v}\right]_q \\ &= \left[\left(\mathbf{A} \cdot Rot(\mathbf{g}\sum_{i=1}^{N} r_i \mathbf{s}_i \mathbf{s}) + \sum_{i=1}^{N} r_i (\mathbf{F}_i Rot(\mathbf{s}) + \mathbf{D}_i \mathbf{E})\right) \cdot \mathbf{v}\right]_q \\ &= \left[\left(\mathbf{A} \cdot Rot(\mathbf{g}\mathbf{s}') + \mathbf{E}'\right) \cdot \mathbf{v}\right]_q \end{aligned}.$$

Since the number of $\mathbf{g}$ in $Rot(\mathbf{g}\mathbf{s}')$ is 1, the encoding $\mathbf{U}$ is a level-1 encoding. □

Lemma 3.4 The encoding $\mathbf{U} \leftarrow \mathrm{Add}\left(\mathrm{par}, k, \mathbf{U}_1 \cdots, \mathbf{U}_t\right)$ is a level-k encoding.

Proof. Let $\mathbf{c}=\left[(\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right]_q$ and $\mathbf{U}_l\cdot\mathbf{A}=\left(\mathbf{A}\cdot Rot(\mathbf{g}^k\mathbf{s}_l)+\mathbf{E}_l\right)\bmod q$.

By $\mathbf{U}=\sum_{l=1}^{t}\mathbf{U}_l$, then

$$\begin{aligned}\left[\mathbf{U}\cdot\mathbf{c}\right]_q&=\left[\sum_{l=1}^{t}\mathbf{U}_l\cdot\mathbf{c}\right]_q\\&=\left[\left(\mathbf{A}\cdot Rot(\mathbf{g}^k\sum_{l=1}^{t}\mathbf{s}_l\mathbf{s})+\sum_{l=1}^{t}(\mathbf{E}_l Rot(\mathbf{s})+\mathbf{U}_l\mathbf{E})\right)\cdot\mathbf{v}\right]_q.\\&=\left[\left(\mathbf{A}\cdot Rot(\mathbf{g}^k\mathbf{s}')+\mathbf{E}'\right)\cdot\mathbf{v}\right]_q\end{aligned}$$

That is, $\mathbf{U}$ is a level-k encoding. □

Lemma 3.5 The encoding $\mathbf{U}\leftarrow\mathrm{Mul}\left(\mathrm{par},1,\mathbf{U}_1,\cdots,\mathbf{U}_k\right)$ is a level-k encoding.

Proof. Let $\mathbf{c}=\left((\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right)\bmod q$ and $\mathbf{U}_l\cdot\mathbf{A}=(\mathbf{A}\cdot Rot(\mathbf{g}\mathbf{s}_l)+\mathbf{E}_l)\bmod q$.

By $\mathbf{U}=\prod_{l=1}^{k}\mathbf{U}_l$, we arrange $\left[\mathbf{U}\cdot\mathbf{c}\right]_q$ to obtain

$$\left[\mathbf{U}\cdot\mathbf{c}\right]_q=\left[\prod_{l=1}^{k}\mathbf{U}_l\cdot\mathbf{c}\right]_q=\left[\left(\mathbf{A}\cdot Rot(\mathbf{g}^k\mathbf{s}\prod_{l=1}^{k}\mathbf{s}_l)+\mathbf{E}'\right)\cdot\mathbf{v}\right]_q.$$

So, $\mathbf{U}$ is a level-k encoding. □

Lemma 3.6 The zero-testing procedure $\mathrm{isZero}\left(\mathrm{par},\mathbf{U},\mathbf{r}\right)$ correctly determines whether $\mathbf{U}$ is a level-d encoding of zero or not.

Proof. By $\mathbf{r}\leftarrow D_{\mathbb{Z}^N,\sigma}$, then $\mathbf{c}=\left[\sum_{i=1}^{N}r_i\cdot\mathbf{c}_i\right]_q=\left[(\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right]_q$.

(1) If $\mathbf{U}$ is a level-d encoding of zero, then $\left[\mathbf{U}\cdot\mathbf{A}\right]_q=\mathbf{F}$ such that $\|\mathbf{F}\|$ is small enough. So,

$$\mathbf{w}=\left[\mathbf{U}\cdot\mathbf{c}\right]_q=\left[\mathbf{U}(\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right]_q=\left[(\mathbf{F}\cdot Rot(\mathbf{s})+\mathbf{U}\mathbf{E})\cdot\mathbf{v}\right]_q.$$

According to our parameters setting, we have $\|\mathbf{w}\|<q^{3/4}$.

(2) If $\mathbf{U}$ is a level-d encoding of non-zero, then $\left[\mathbf{U}\cdot\mathbf{A}\right]_q=\left[\mathbf{A}\cdot Rot(\mathbf{g}^d\mathbf{s}')+\mathbf{F}\right]_q$. Hence

$$\mathbf{w}=\left[\mathbf{U}\cdot\mathbf{c}\right]_q=\left[\mathbf{U}(\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right]_q=\left[(\mathbf{A}\cdot Rot(\mathbf{g}^d\mathbf{s}')+\mathbf{F}\cdot Rot(\mathbf{s})+\mathbf{U}\mathbf{E})\cdot\mathbf{v}\right]_q.$$

By $\|(\mathbf{F}\cdot Rot(\mathbf{s})+\mathbf{U}\mathbf{E})\cdot\mathbf{v}\|<q^{3/4}$, $\|\mathbf{A}\|\approx q$ and $Rot(\mathbf{g}^d\mathbf{s}')\cdot\mathbf{v}\neq\mathbf{0}$, we get $\left\|\mathbf{A}\cdot Rot(\mathbf{g}^d\mathbf{s}')\cdot\mathbf{v}\right\|\approx q$ with overwhelming probability.

Thus, isZero can correctly decide whether $\mathbf{U}$ is a level-d encoding of zero. □

Lemma 3.7 Suppose that two level-d encodings $\mathbf{U}_1,\mathbf{U}_2$ encode same plaintext. Then

$$\mathrm{Ext}\left(\mathrm{par},\mathbf{U}_1,\mathbf{r}\right)=\mathrm{Ext}\left(\mathrm{par},\mathbf{U}_2,\mathbf{r}\right).$$

Proof. Since $\mathbf{U}_j$, $j\in[\![2]\!]$ encode same plaintext, then $\left[\mathbf{U}_j\cdot\mathbf{A}\right]_q=\left[\mathbf{A}\cdot Rot(\mathbf{g}^d\mathbf{s}')+\mathbf{F}_j\right]_q$.

By $\mathbf{r}\leftarrow D_{\mathbb{Z}^N,\sigma}$, then $\mathbf{c}=\left[\sum_{i=1}^{N}r_i\cdot\mathbf{c}_i\right]_q=\left[(\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right]_q$. Therefore

$$\begin{aligned}\mathbf{w}_j&=\left[\mathbf{U}_j\cdot\mathbf{c}\right]_q\\&=\left[\mathbf{U}_j(\mathbf{A}\cdot Rot(\mathbf{s})+\mathbf{E})\cdot\mathbf{v}\right]_q\\&=\left[(\mathbf{A}\cdot Rot(\mathbf{g}^d\mathbf{s}')+\mathbf{F}_j\cdot Rot(\mathbf{s})+\mathbf{U}_j\mathbf{E})\cdot\mathbf{v}\right]_q\\&=\left[\mathbf{A}\cdot Rot(\mathbf{g}^d\mathbf{s}')\cdot\mathbf{v}+(\mathbf{F}_j\cdot Rot(\mathbf{s})+\mathbf{U}_j\mathbf{E})\cdot\mathbf{v}\right]_q\end{aligned}$$

Since $\left\|(\mathbf{F}_j \cdot Rot(\mathbf{s}) + \mathbf{U}_j \mathbf{E}) \cdot \mathbf{v}\right\| < q^{3/4}$ and $\left\|\mathbf{A} \cdot Rot(\mathbf{g}^d \mathbf{s}') \cdot \mathbf{v}\right\| \approx q$ with overwhelming probability.

Hence, the v most-significant bits of each entry of $\mathbf{w}_1$ are equal to that of $\mathbf{w}_2$. □

3.3 Hardness assumption

In our variation, one can determine whether an encoding encodes zero or not when its level is not greater than d. So, we adaptively define hardness assumptions for our variation to contain this situation.

Consider the following security experiment:

(1) $\text{par} \leftarrow \text{InstGen}(1^\lambda, 1^d)$

(2) Choose an integer $0 < k \le d$

(2) For $l = 0$ to k:

Sample $\mathbf{t}_l \leftarrow D_{\mathbb{Z}^N, \sigma}$;

Generate level-1 encoding $\mathbf{U}_l = \sum_{i=1}^{N} t_{l,i} \cdot \mathbf{D}_i$.

(3) Set $\mathbf{U} = \prod_{l=1}^{k} \mathbf{U}_l$.

(4) Set $\mathbf{w}_C = \mathbf{w}_D = \text{Ext}\left(\text{par}, \mathbf{U}, \mathbf{t}_0\right)$.

(5) Set $\mathbf{w}_R = \text{Ext}\left(\text{par}, \mathbf{U}, \mathbf{r}_0\right)$ with $\mathbf{r}_0 \leftarrow D_{\mathbb{Z}^N, \sigma}$.

Definition 3.8 (ext-GCDH/ext-GDDH). According to the security experiment, the ext-GCDH and ext-GDDH are defined as follows:

Level-k extraction CDH (ext-GCDH): Given $\left\{\text{par}, \mathbf{U}_0, \cdots, \mathbf{U}_k\right\}$, output a level-$k$ extraction encoding $\mathbf{w} \in \mathbb{Z}_q^m$ such that $\mathbf{w}_C = \text{Extract}_s\left(\text{msbs}_v\left(\mathbf{w}\right)\right)$.

Level-k extraction DDH (ext-GDDH): Given $\left\{\text{par}, \mathbf{U}_0, \cdots, \mathbf{U}_k, \mathbf{w}\right\}$, distinguish between $D_{ext-GDDH} = \left\{\text{par}, \mathbf{U}_0, \cdots, \mathbf{U}_k, \mathbf{w}_D\right\}$ and $D_{ext-RAND} = \left\{\text{par}, \mathbf{U}_0, \cdots, \mathbf{U}_k, \mathbf{w}_R\right\}$.

In this paper, we assume that the ext-GCDH/ext-GDDH is hard.

4 Cryptanalysis

In this section, we first give easily computable some quantities in our variation, then analyze that Coron attack [Cor15] does not work for our construction using these quantities.

4.1 Easily computable quantities

Given the public parameters par, we can generate special hidden encodings of zero using cross multiplication. That is, we can compute some quantities that are not reduced over modulo. In the following, we give some easily computable quantities.

Given $\text{par} = \left\{q, \left\{\mathbf{D}_i, \mathbf{c}_i\right\}_{i \in [\![N]\!]}\right\}$, then for $i, j \in [\![N]\!]$ and $i \neq j$, we have

$$\begin{aligned}\mathbf{w}_{i,j} &= \left[\mathbf{D}_i \cdot \mathbf{c}_j - \mathbf{D}_j \cdot \mathbf{c}_i\right]_q \\ &= \left[(\mathbf{D}_i \cdot \mathbf{C}_j - \mathbf{D}_j \cdot \mathbf{C}_i) \cdot \mathbf{v}\right]_q \\ &= \left[(\mathbf{D}_i \cdot (\mathbf{A} \cdot Rot(\mathbf{s}_j) + \mathbf{E}_j) - \mathbf{D}_j \cdot (\mathbf{A} \cdot Rot(\mathbf{s}_i) + \mathbf{E}_i)) \cdot \mathbf{v}\right]_q . \\ &= \left[(\mathbf{F}_i \cdot Rot(\mathbf{s}_j) + \mathbf{D}_i\mathbf{E}_j - \mathbf{F}_j \cdot Rot(\mathbf{s}_i) - \mathbf{D}_j\mathbf{E}_i) \cdot \mathbf{v}\right]_q \\ &= (\mathbf{F}_i \cdot Rot(\mathbf{s}_j) + \mathbf{D}_i\mathbf{E}_j - \mathbf{F}_j \cdot Rot(\mathbf{s}_i) - \mathbf{D}_j\mathbf{E}_i) \cdot \mathbf{v}\end{aligned}$$

According to our parameters setting, it is easy to verify that $\mathbf{w}_{i,j}$ is not reduced modulo q. Moreover, we can also compute similarly non-reduced quantities using addition and multiplication of encodings. However, at present we do not know efficient attacks using these non-reduced quantities $\mathbf{w}_{i,j}$.

4.2 Coron Attack

Recently, Coron [Cor15] described an attack of multipartite key exchange protocol using GGH15. The Coron's attack consists of two steps: (1) The first step represents one secret plaintext s_1 as a linear combination of the other secret plaintexts $t_{1,l}$ using a variant of the Cheon et al. attack; (2) The second step generates an equivalent private encoding by using the previous linear combination.

Similar to the first step of the Coron's attack, we can build a linear relation between secret plaintexts. According to the encoding $\mathbf{U} = \sum_{i=1}^{N} r_i \cdot \mathbf{D}_i$, we can compute α_i's using LLL algorithm such that $\mathbf{U} = \sum_{i=1}^{N} \alpha_i \cdot \mathbf{D}_i$. However, we cannot find small integers α_i's.

However, we cannot generate an equivalent encoding as that in [Cor15]. In the second step, Coron computes an equivalent private key by using more than 2 public encodings to solve the problem of large integers α_i's . Namely, the Coron's attack is only applicable for 3 users or more. However, the public parameters in our construction cannot satisfy this condition. So, our variation seems to thwart the Coron's attack.

5 Applications

In this section, we construct multipartite key exchange protocol and witness encryption using our variation.

5.1 MPKE Protocol

We describe a one-round multipartite Diffie-Hellman key exchange protocol. The security relies on the hardness assumption of ext-GDDH.

Setup$(1^\lambda, 1^{d+1})$. Output $(\text{par}) \leftarrow \text{InstGen}(1^\lambda, 1^d)$ as the public parameters.

Publish(par, j) . The j -th party samples $\mathbf{t}_j \leftarrow D_{\mathbb{Z}^N,\sigma}$, publishes the public key $\mathbf{U}_j = \sum_{i=1}^{N} t_{j,i} \cdot \mathbf{D}_i$ and remains $\mathbf{t}_j$ as the secret key.

KeyGen$\left(\text{par}, j, \mathbf{t}_j, \{\mathbf{U}_l\}_{l \neq j}\right)$. The j -th party extracts the common secret key $sk = \text{Ext}\left(\text{par}, \mathbf{U}'_j, \mathbf{t}_j\right)$, where $\mathbf{U}'_j = \prod_{l \neq j} \mathbf{U}_l$.

Theorem 5.1 Suppose that the ext-GCDH/ext-GDDH defined in Section 3.3 is hard, then our

construction is one-round multipartite Diffie-Hellman key exchange protocol.

5.2 Witness Encryption

Garg, Gentry, Sahai, and Waters [GGSW13] constructed an instance of witness encryption based on the NP-complete 3-exact cover problem and the GGH map. However, Hu and Jia [HJ15a] have broken the GGH-based WE. Here we describe a witness encryption using our variation.

3-Exact Cover Problem [GGH13, Gol08] Given a collection S of subsets $T_1, T_2, ..., T_\pi$ of $[\![\tau]\!] = \{1, 2, ..., \tau\}$ such that $\tau = 3\theta$ is a multiple of 3 and $|T_i| = 3$, find a 3-exact cover of $[\![\tau]\!]$. For an instance of witness encryption, the public key is the public parameters $\text{par} = \left\{q, \{\mathbf{D}_i, \mathbf{c}_i\}_{i \in [\![N]\!]}\right\}$ and a collection S, and the secret key is a hidden 3-exact cover of $[\![\tau]\!]$.

Encrypt(M)**:**

(1) For $j \in [\![\tau]\!]$, sample $\mathbf{t}_j \leftarrow D_{\mathbb{Z}^N, \sigma}$ and generate level-1 encodings $\mathbf{U}_j = \sum_{i=1}^{N} t_{j,i} \cdot \mathbf{D}_i$.

(2) Compute $\mathbf{U} = \prod_{j=1}^{\tau} \mathbf{U}_j$ and $sk = \text{Ext}(\text{par}, \mathbf{U}, \mathbf{e})$, and encrypt a message M into ciphertext C using the secret key sk, where $\mathbf{e} = (1, 0, ..., 0) \in \mathbb{Z}^N$.

(3) For each element $T_i = \{i_1, i_2, i_3\}$, generate a level-3 encoding $\mathbf{U}_{T_i} = \mathbf{U}_{i_1} \mathbf{U}_{i_2} \mathbf{U}_{i_3}$. Let $E = \left(\mathbf{U}_{T_i}, T_i \in S\right)$.

(4) Output the ciphertext $CT = (C, E)$.

Decrypt(CT, W):

(1) Given CT and a witness set W, compute $\mathbf{U} = \prod_{T_i \in W} \mathbf{U}_{T_i}$.

(2) Generate $sk = \text{Ext}(\text{par}, \mathbf{U}, \mathbf{e})$, and decrypt C to a message M.

Similar to [GGSW13a], the security of our construction depends on the hardness assumption of the Decision Graded Encoding No-Exact-Cover.

Theorem 5.2 Suppose that the Decision Graded Encoding No-Exact-Cover is hard. Then our construction is a witness encryption scheme.

We observe that the Hu-Jia attack [HJ15b] does not work for our construction. This is because that matrix does not support commutative law. Roughly speaking, even if a combined element $T_i = T_j \bigcup T_k - T_l$, the encoding $\mathbf{U}_{T_i} \neq \mathbf{U}_{T_j} \cdot \mathbf{U}_{T_k} \cdot (\mathbf{U}_{T_l})^{-1}$.

References

[BGH+15] Z. Brakerski, C. Gentry, S. Halevi, T. Lepoint, A. Sahai, M. Tibouchi. Cryptanalysis of the Quadratic Zero-Testing of GGH. http://eprint.iacr.org/2015/845.

[BR14] Z. Brakerski and G. N. Rothblum. Virtual black-box obfuscation for all circuits via generic graded encoding. TCC 2014, LNCS 8349, pp. 1-25.

[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.

[BZ14] D. Boneh and M. Zhandry. Multiparty key exchange, efficient traitor tracing, and more from indistinguishability obfuscation. CRYPTO 2014, LNCS 8616, pp. 480-499.

[CGH+15] J. S. Coron, C. Gentry, S. Halevi, T. Lepoint, H. K. Maji, E. Miles, M. Raykova, A. Sahai, M. Tibouchi. Zeroizing Without Low-Level Zeroes New MMAP Attacks and Their Limitations. http://eprint.iacr.org/2015/596.

[CHL+15] 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.

[CL15] J. H. Cheon, C. Lee. Cryptanalysis of the multilinear map on the ideal lattices. http://eprint.iacr.org/2015/461.

[CLR15] J. H. Cheon, C. Lee, H. Ryu. Cryptanalysis of the New CLT Multilinear Maps. http://eprint.iacr.org/2015/934.

[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.

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

[Cor15] J. S. Coron. Cryptanalysis of GGH15 Multilinear Maps. http://eprint.iacr.org/2015/1037.

[GG13] J. von zur Gathen, J. Gerhard. Modern computer algebra [M]. 3rd edition, Cambridge: Cambridge University Press, 2013.

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

[GGH+13a] S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B. Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. FOCS 2013, pp.40-49.

[GGH+13b] S. Garg, C. Gentry, S. Halevi, A. Sahai, and B. Waters. Attribute-based encryption for circuits from multilinear maps, CRYPTO (2) 2013, LNCS 8043, 479-499.

[GGH+14] S. Garg, C. Gentry, S. Halevi, and M. Zhandry. Fully secure functional encryption without obfuscation. http://eprint.iacr.org/2014/666.

[GGH15] C. Gentry, S. Gorbunov, and S. Halevi. Graph-induced multilinear maps from lattices. TCC 2015, Part II, LNCS 9015, pp. 498–527.

[GHM+14] C. Gentry, S. Halevi, H. K. Majiy, A. Sahaiz. Zeroizing without zeroes: Cryptanalyzing multilinear maps without encodings of zero. http://eprint.iacr.org/2014/929.

[Gol08] O. Goldreich. Computational Complexity: a Conceptual Perspective. Cambridge University Press, New York, NY, USA, 1 edition, 2008.

[GGSW13] S. Garg, C. Gentry, A. Sahai, and B. Waters. Witness encryption and its applications. STOC 2013, pp. 467-476.

[GSW13] C. Gentry, A. Sahai and B. Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. CRYPTO (1) 2013, LNCS 8042, pp. 75-92.

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

[Hal15] Shai Halevi. The state of cryptographic multilinear maps, 2015. Invited Talk of CRYPTO 2015.

[HIL+99] J. Hastad, R. Impagliazzo, L. A. Levin, and M. Luby. A pseudorandom generator from any one-way function. SIAM Journal on Computing, 1999, 28(4):1364-1396.

[HJ15a] Yupu Hu and Huiwen Jia. Cryptanalysis of GGH Map. http://eprint.iacr.org/2015/301.

[HJ15b] Yupu Hu and Huiwen Jia. A Comment on Gu Map-1. http://eprint.iacr.org/2015/448.

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

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

[MF15] B. Minaud and P. Fouque. Cryptanalysis of the New Multilinear Map Over the integers, http://eprint.iacr.org/2015/941.