An Optimization of Gu Map-1*

Yupu Hu and Huiwen Jia
ISN Laboratory, Xidian University, 710071 Xi’an, China
yphu@mail.xidian.edu.cn

Abstract. As a modified version of GGH map, Gu map-1 was successful in constructing multi-party key exchange (MPKE). In this short paper we present a result about the parameter setting of Gu map-1, therefore we can reduce a key parameter \( \tau \) from original \( O(n^2) \) down to \( O(\lambda n) \) (in theoretically secure case, where \( \lambda \) is the security parameter), and even down to \( O(2n) \) (in computationally secure case). Such optimization greatly reduces the size of the map.

Keywords: Multilinear maps, GGH map, Gu map-1, Multi-party key exchange (MPKE), Lattice based cryptography.

1 Introduction: Background and Our Comment

Because we presented efficient attack [1] on GGH map for given encodings of zero [2, 3], modification of GGH map is urgently needed. Gu map-1 [4] is one of modified versions of GGH map. It successfully forms MPKE scheme, and avoids our attack.

In this short paper we present a result about the parameter setting of Gu map-1, therefore we can reduce a key parameter \( \tau \) from original \( O(n^2) \) down to \( O(\lambda n) \) (in theoretically secure case, where \( \lambda \) is the security parameter), and even down to \( O(2n) \) (in computationally secure case). Our result is that “vector group” \( \{Y_i, i = 1, \cdots, \tau\} \) has the “rank” \( n \) rather than \( n^2 \), and that “vector group” \( \{P_{xt,i}, i = 1, \cdots, \tau\} \) has the “rank” \( n \) rather than \( n^2 \). Such optimization greatly reduces the size of the map.

2 Gu Map-1

2.1 Setting Parameters

We define the integers by \( \mathbb{Z} \). We specify that \( n \)-dimensional vectors of \( \mathbb{Z}^n \) are row vectors. We consider the \( 2n \)'th cyclotomic polynomial ring \( R = \mathbb{Z}[X]/(X^n + 1) \), and identify an element \( u \in R \) with the coefficient vector of the degree-\((n - 1)\) integer polynomial that represents \( u \). In this way, \( R \) is identified with the integer lattice \( \mathbb{Z}^n \). We also consider the ring \( R_q = R/qR = \mathbb{Z}_q[X]/(X^n + 1) \) for a (large

* The work was supported in part by Natural Science Foundation of China under Grant 60833008
Choose a prime $q$. Obviously, addition in these rings is done component-wise in their coefficients, and multiplication is polynomial multiplication modulo the ring polynomial $X^n + 1$. For $u \in R$, we denote $Rot(u) = \begin{bmatrix} u_0 & u_1 & \cdots & u_{n-1} \\ -u_{n-1} & u_0 & \cdots & u_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ -u_1 & -u_2 & \cdots & u_0 \end{bmatrix} \in \mathbb{Z}_{n \times n}$.

Because Gu map-1 scheme uses the GGH construction [2, 3] as the basic component, parameter setting is set as that of GGH to conveniently describe and compare. Let $\lambda$ be the security parameter, $K$ the multilinearity level, $n$ the dimension of elements of $R$. Concrete parameters are set as $\sigma = \sqrt{n}$, $\sigma' = \lambda n^{1.5}$, $\sigma^* = 2^\lambda$, $q \geq 2^{8K\lambda n^{O(K)}}$, $n \geq O(K^2)$, $\tau = O(n^2)$. More detailed setting of $\tau$ is $\tau = n^2 + \lambda$.

### 2.2 Instance Generation

1. Choose a prime $q \geq 2^{8K\lambda n^{O(K)}}$.
2. Choose an element $g \leftarrow D_{\mathbb{Z}^n, \sigma}$ in $R$ so that $\|g^{-1}\| \leq n^2$. In other words, $g$ is “very small”.
3. Choose elements $a_i, e_i \leftarrow D_{\mathbb{Z}^n, \sigma}$, $b_i \leftarrow D_{\mathbb{Z}^\sqrt{q}, \sigma}$, $i = 1, \cdots, \tau$ in $R$. In other words, $a_i, e_i$ are “very small”, while $b_i$ is “somewhat small”.
4. Choose a random element $z \in R_q$. In other words, $z \in R_q$ is never small.
5. Choose two matrices $T, S \leftarrow D_{\mathbb{Z}^{n \times n}, \sigma}$. In other words, $T$ and $S$ are “very small”.
6. Set $Y_i = \left[ T \text{Rot}(\frac{a_i g + e_i}{z})T^{-1} \right]_q$, $P_{z,t,i} = \left[ T \text{Rot}(\frac{z^K(b_i g + e_i)}{q})S \right]_q$, $i = 1, \cdots, \tau$.
7. Output the public parameters $\{q, \{Y_i, P_{z,t,i}\}, i = 1, \cdots, \tau \}$.
8. Generating level-1 encodings. A user generates his secret $d \leftarrow D_{\mathbb{Z}^n, \sigma^*}$ in $R$, then publishes $U = \left[ \sum_{i=1}^{\tau} d_i Y_i \right]_q = \left[ T \text{Rot}(\frac{\sum_{i=1}^{\tau} d_i(a_i g + e_i)}{z})T^{-1} \right]_q$. $U$ is level-1 encoding of the secret $d$.
9. Generating level-$K$ decoding factors. After the user generating his secret $d$, he secretly computes $V = \left[ \sum_{i=1}^{\tau} d_i P_{z,t,i} \right]_q = \left[ T \text{Rot}(\frac{z^K \sum_{i=1}^{\tau} d_i(b_i g + e_i)}{q})S \right]_q$. $V$ is level-$K$ decoding factor of the secret $d$.

### 2.3 A Note

Each public matrix $Y_i$ has $n^2$ entries, and so does $P_{z,t,i}$. For a public encoding $U = \left[ \sum_{i=1}^{\tau} d_i Y_i \right]_q$, if $\{Y_i, i = 1, \cdots, \tau \}$ are linearly independent, the secret $\{d_i, i = 1, \cdots, \tau \}$ can be uniquely solved. This is the reason that $\tau > n^2$, and more detailed that $\tau = n^2 + \lambda$. 
3 A Result about Parameter Setting of Gu Map-1

3.1 Our Notations

We denote \( \begin{bmatrix} u_{i,0} & u_{i,1} & \cdots & u_{i,n-1} \end{bmatrix}_q \). We denote \( T = \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_n \end{bmatrix} \), where \( T_k \) is the \( k \)'th row of \( T \). We denote \( T^{-1} = \begin{bmatrix} T_1^{-1} & T_2^{-1} & \cdots & T_n^{-1} \end{bmatrix} \), where \( T_k^{-1} \) is the \( k \)'th column of \( T^{-1} \). For \( T_k^{-1}(l) = \begin{bmatrix} t_{k,1}^{-1} \\ t_{k,2}^{-1} \\ \vdots \\ t_{k,n}^{-1} \end{bmatrix} \), we define \( T_k^{-1}(l) = \begin{bmatrix} t_{k,1}^{-1} \\ t_{k,2}^{-1} \\ \vdots \\ t_{k,n}^{-1} \end{bmatrix} \).

3.2 A Result about Vector \( Y_i \)

Proposition 1

(1) \( e_{Y_i} = \begin{bmatrix} u_{i,0} \\ u_{i,1} \\ \vdots \\ u_{i,n-1} \end{bmatrix}_q A \), where \( A \) is \( n \times (n^2) \) matrix, which is only dependent on \( T \).

(2) We denote \( l \) as the serial numbers of rows of \( A \), \( l = 0, 1, \cdots, n - 1 \). We denote \( (k, j) \) as the serial numbers of columns of \( A \), \( k, j = 1, 2, \cdots, n \). Then \( (l, (k, j)) \) entry of \( A \) is \( T_k T_j^{-1}(l) \).

(3) Special entries of \( A \). \( (0, (k, j)) \) entry of \( A \) is 1 for \( k = j \), and 0 for \( k \neq j \).

(4) A natural corollary. The rank of \( \{ e_{Y_i}; i = 1, \cdots, \tau \} \) is at most \( n \) rather than \( n^2 \).

3.3 Similar Result about Vector \( P_{zt,i} \)

Proposition 2 We write matrix \( P_{zt,i} \) into the form of “vector” \( \tilde{P}_{zt,i} \). Then the rank of \( \{ \tilde{P}_{zt,i}; i = 1, \cdots, \tau \} \) is at most \( n \) rather than \( n^2 \).
4 Reducing $\tau$

Suppose we obtain a public encoding $U = \left[ \sum_{i=1}^{\tau} d_i Y_i \right]_q$.

$\tau > n$ will guarantee $\{Y_i, i = 1, \cdots, \tau\}$ linearly dependent, therefore the secret $\{d_i, i = 1, \cdots, \tau\}$ can not be uniquely solved.

$\tau = \lambda n$ will guarantee that SVP (the shortest vector problem) over the lattice generated by $\{Y_i, i = 1, \cdots, \tau\}$ is theoretically hard. Notice that $\lambda$ is far smaller than $n$.

$\tau = 2n$ will guarantee that SVP over the lattice generated by $\{Y_i, i = 1, \cdots, \tau\}$ is computationally hard.

References