A Method to Reduce the Key Size of UOV Signature Scheme

Chengdong Tao

Beijing Institute of Mathematical Sciences and Applications, Beijing, China
taochengdong@bimsa.cn

Abstract. Multivariate public key signature scheme has a good performance on speed and signature size. But most of them have a huge public key size. In this paper, we propose a new method to reduce the public key size of unbalance oil and vinegar (UOV) signature scheme. We can reduce the public key size of UOV scheme to about 4KB for 128 bits security level. This method can be used to reduce the public key sizes of other multivariate public key cryptosystems.

Keywords: Post quantum cryptography · Multivariate public key cryptography · UOV signature scheme · Circulant matrix · Toeplitz matrix..

1 Introduction

Multivariate public key cryptosystem is one of the main candidates for post quantum cryptosystem. Multivariate public key cryptography is one of the major candidate for post quantum cryptography. It's security base on the hardness of solving multivariate polynomials equations over a finite field. A problem of solving a system of multivariate polynomials equations whose degrees are no less than 2 over a finite field is called multivariate polynomials (MP for short) problem. The MP problem is proved to be NP-complete. More precisely, MP problem is equivalent to the 3-SAT problem [1]. In practically, most of multivariate public key cryptosystems (MPKC for short) are restricted to multivariate quadratic polynomials because of efficiency. A problem of solving a system of multivariate quadratic (MQ for short) polynomials equations over a finite field is called MQ problem. The MQ problem is also proved to be NP-complete [1].

In the last three decades, many multivariate public key signature schemes have been proposed but most of them are broken, except some UOV-based (such as UOV [2] and Rainbow [3][3]) signature schemes. Moreover, Rainbow which is UOV-based signature scheme is acceptance of the third round of National Institute of Standards and Technology Post-Quantum Cryptography Projects for signature.

In 1997, Patarin firstly used oil-vinegar polynomials to build a signature scheme, namely oil and vinegar signature scheme(OV scheme) [7]. But in 1998, Kipnis and Shamir showed that the OV scheme is insecure [8].

In 1999, Kipnis et al. improved the OV scheme and proposed Unbalance Oil and Vinegar scheme (UOV scheme)[2]. However, the security UOV scheme has

at lest 3 times more variables than polynomials, this means that signature size of UOV is at least 3 times longer than hash value of document. Moreover, the public key size of UOV scheme is too large.

In order to improving the efficiency of UOV scheme, Ding and Schmidt proposed Rainbow signature scheme, which is a multi-layer construction using unbalance oil and vinegar polynomial at each layer. Rainbow scheme can generate a more shorter signature than UOV scheme, but the public key size is still very large.

In order to reducing the public key size of UOV, some methods have been tried. In [16], Beullens et al. used field lifting method to reduce the UOV public key size and obtain a small public key size, namely LUOV signature scheme, but the signature size becomes more longer than the origin UOV scheme on the same security level. However, LUOV was broken by Jintai Ding et al.[23]. In [18,19], Petzold et al. used linear recurring sequences to reduce the public key size, they managed to reduce the key size by a factor of 8 in the case of UOV and a factor of 3 in the case of the Rainbow signature scheme without expanding the signature size on the same security level.

In this paper, we propose a new method to reduce the public key size of UOV scheme. We can reduce the public key size to 4.096KB for 128 bits security level. This method can be used to reduce the public key size of other multivariate public key cryptosystems.

In order to distinguish the original UOV signature scheme, the optimized UOV scheme by using our method are called Hufu[1]-UOV signature scheme. We place all software described in this paper into the public domain and make it available online at `https://github.com/hufuuov/hufu_uov.git`.

The paper is organized as follows. In Sect.2, we introduce general multivariate public key signature scheme. In Sect.3, we introduce the original UOV signature scheme. In Sect.4, we introduce Hufu-UOV signature scheme. In Sect.5, we present cryptanalysis of Hufu-UOV scheme. In Sect.6, we proposed some parameters for Hufu-UOV . Finally, we conclude the paper in Sect.7.

2 General Multivariate Public Key Signature Scheme

Let $\mathbb{N}$ be a set of positive integers, $m, n, q \in \mathbb{N}$, and $\mathbb{F}_q$ be a finite field with q elements. Let $\mathbb{F}_q[x_1, x_2, \cdots, x_n]$ be a polynomial ring over $\mathbb{F}_q$. In generally, the public key of a multivariate public key cryptosystem (MPKC for short) is a set of multivariate quadratic polynomials:

$$\left\{p^{(1)}(x_1, x_2, \cdots, x_n), \cdots, p^{(m)}(x_1, x_2, \cdots, x_n)\right\},$$

[1] The Hufu is a tiger-shaped metal military vouchers of Ancient China. It was divided into two halves and issued by the emperor to the general. The right half is held by the emperor, the left half is sent to the general. Each army has a corresponding Hufu. When the troops are mobilized, the two halves must be combined, if they match, then the general accepts the command, otherwise, rejects the command. This is similar to signature and verification. So we call our new signature scheme to Hufu.

or equivalently, a multivariate quadratic map:

$$\mathcal{P} : \mathbb{F}_q^n \mapsto \mathbb{F}_q^m$$
$$\mathcal{P} = (p^{(1)}(x_1, x_2, \cdots, x_n), \cdots, p^{(m)}(x_1, x_2, \cdots, x_n)),$$

where

$$p^{(k)} = \sum_{1 \le i \le j \le n} a_{ij}^{(k)} x_i x_j + \sum_{i=1}^{n} b_i^{(k)} x_i + c^{(k)},$$

$a_{ij}^{(k)}, b_i^{(k)}, c^{(k)} \in \mathbb{F}_q, (k = 1, \cdots, m)$.

To build a MPKC, we start to build a trapdoor function (central map): $\mathcal{F} : \mathbb{F}_q^n \mapsto \mathbb{F}_q^m$. In generally, central map is a quadratic multivariate map, and easy to inverse. In order to hide the structure of central map in the public key, we compose it with two invertible linear or affine transformations $\mathcal{L}_1 : \mathbb{F}_q^n \mapsto \mathbb{F}_q^n$ and $\mathcal{L}_2 : \mathbb{F}_q^m \mapsto \mathbb{F}_q^m$.

Key generation. The public key is quadratic multivariate map: $\mathcal{P} = \mathcal{L}_2 \circ \mathcal{F} \circ \mathcal{L}_1$. The private key is $\mathcal{L}_2^{-1}$, $\mathcal{F}$ and $\mathcal{L}_1^{-1}$. In generally, the public key size of multivariate public key cryptosystem is

$$\text{Size}_{\text{pk}} = \begin{cases} m \cdot \frac{(n+1)(n+2)}{2} & \text{if } q \neq 2, \\ m \cdot \left(\frac{n(n+1)}{2} + 1 \right) & \text{if } q = 2, \end{cases}$$

field elements. The private key can be generated by a randomly choosing seed, therefore the private key size is equal to the size of seed.

Signature generation. Let $\mathbf{m}$ be a document to be signed, $\mathbf{h} = hash(\mathbf{m}) \in \mathbb{F}_q^m$ be the hash value of $\mathbf{m}$. The signer computes recursively $\mathbf{y} = \mathcal{L}_2^{-1}(\mathbf{h})$, $\mathbf{z} = \mathcal{F}^{-1}(\mathbf{y})$, and $\mathbf{s} = \mathcal{L}_1^{-1}(\mathbf{z})$. Then $\mathbf{s}$ is the signature of document $\mathbf{m}$.

Signature verification. To verify the $\mathbf{s}$ is indeed a valid signature of document $\mathbf{m}$, the recipient computes $\mathbf{h} = hash(\mathbf{m})$ and $\mathcal{P}(\mathbf{s})$. If $\mathcal{P}(\mathbf{s}) = \mathbf{h}$, the recipient accepts signature, otherwise the recipient rejects signature.

3 Original UOV Signature Scheme

Let o, v be positive integers, and $n = o + v, m = o$. The central map of UOV signature scheme is built by some oil and vinegar polynomials which are defined as follow:

Definition 1. *An oil and vinegar polynomial is any total degree two polynomial $f \in \mathbb{F}_q[x_1, x_2, \cdots, x_n]$ of the form*

$$f(x_1, x_2, \cdots, x_n) = \sum_{i=o+1}^{n} \sum_{j=1}^{n} a_{ij} x_i x_j + \sum_{i=1}^{n} b_i x_i + c,$$

where $a_{ij}, b_i, c \in \mathbb{F}_q$.

The variables $x_1, \cdots, x_o$ are called the oil variables, and the variables $x_{o+1}, \cdots, x_n$ are called the vinegar variables. If we fix the values of vinegar variables, the oil and vinegar polynomial will become a linear polynomial.

Key generation. The polynomials in the central map of UOV scheme are the oil and vinegar polynomials, namely the central map

$$\mathcal{F} = (f_1, f_2, \cdots, f_o) : \mathbb{F}_q^n \to \mathbb{F}_q^o,$$

where $f_1, f_2, \cdots, f_o$ are the oil and vinegar polynomials. The public key of UOV scheme is $\mathcal{P} = \mathcal{F} \circ \mathcal{L}_1$, where $\mathcal{L}_1 : \mathbb{F}_q^n \to \mathbb{F}_q^n$ is invertible affine transformation. The private key consists of $\mathcal{F}$ and $\mathcal{L}_1^{-1}$.

Signature generation. Let $\mathbf{m}$ be a document to be signed, $\mathbf{h} = (h_1, h_2, \cdots, h_o) = hash(\mathbf{m}) \in \mathbb{F}_q^o$ be the hash value of $\mathbf{m}$. The process of signature generation is as follows:

1. Compute $\mathcal{F}^{-1}(h_1, h_2, \cdots, h_o)$. By randomly choosing $(\acute{x}_{o+1}, \cdots, \acute{x}_n) \in \mathbb{F}_q^v$ to give the values of the vinegar variables, we obtain a linear system in the oil variables $x_1, x_2, \cdots, x_o$ given by

   $$\mathcal{F}(x_1, x_2, \cdots, x_o, \acute{x}_{o+1}, \cdots, \acute{x}_n) = (h_1, \cdots, h_o).$$

   The probability of this linear system will has a solution is roughly $1 - \frac{1}{q}$. If the linear system has no solution, we choose different values for the vinegar variables, until the system has a solution. Finally, we solve the linear system and obtain a vector $(\acute{x}_1, \cdots, \acute{x}_o, \acute{x}_{o+1}, \cdots, \acute{x}_n) \in \mathbb{F}_q^n$ such that $(\acute{x}_1, \cdots, \acute{x}_o, \acute{x}_{o+1}, \cdots, \acute{x}_n) = \mathcal{F}^{-1}(y_1, y_2, \cdots, y_o)$.
2. We compute

   $$(z_1, z_2, \cdots, z_n) = \mathcal{L}_1^{-1}(\acute{x}_1, \cdots, \acute{x}_o, \acute{x}_{o+1}, \cdots, \acute{x}_n),$$

   then $(z_1, z_2, \cdots, z_n)$ is the signature of document $\mathbf{m}$.

Signature verification. To verify the $(z_1, \cdots, z_n)$ is indeed a valid signature of document $\mathbf{m}$, the recipient computes $\mathbf{h} = hash(\mathbf{m})$ and $\mathcal{P}(z_1, z_2 \cdots, z_n)$. If $\mathcal{P}(z_1, z_2 \cdots, z_n) = \mathbf{h}$, the recipient accepts signature, otherwise the recipient rejects signature.

When $o = v$, the UOV scheme is the original oil and vinegar signature scheme proposed by Patarin[7]. It was called the balanced oil and vinegar scheme and broken by Kipnis and Shamir[8]. When $o < v$, it was called the unbalanced oil and vinegar scheme. When $v < 2o$, the attack complexity is $\mathcal{O}(o^4 q^{n-2o-1})$ by using separation of oil and vinegar variables attack[2]. When $v \geq 2o$, the security of UOV scheme is still an open question. However, when $v \geq 2o$, the UOV scheme has 3 times more variables than polynomials.

3.1 Equivalent Keys of UOV

Let $(\mathcal{P}, (\mathcal{F}, \mathcal{L}_1))$ be a key pair of UOV and $\Omega : \mathbb{F}_q^n \to \mathbb{F}_q^n$ be an invertible affine transformation, then we get

$$\mathcal{P} = \mathcal{F} \circ \mathcal{L}_1 = \mathcal{F} \circ \Omega \circ \Omega^{-1} \circ \mathcal{L}_1.$$

Theorem 1. *Let $(\mathcal{F}, \mathcal{L}_1)$ be a private key of UOV, $\Omega : \mathbb{F}_q^n \to \mathbb{F}_q^n$ be an invertible affine transformation whose linear part has the form*

$$\begin{pmatrix} \Omega_{v\times v}^{(1)} & 0_{v\times o} \\ \Omega_{o\times v}^{(2)} & \Omega_{o\times o}^{(3)} \end{pmatrix}. \tag{1}$$

Then $(\mathcal{F} \circ \Omega, \Omega^{-1} \circ \mathcal{L}_1)$ is an equivalent key of UOV.

Proof. see [20].

Theorem 2. *Let $\mathcal{P}$ be a public key of UOV, then with overwhelming probability, there exists a equivalent key $\tilde{\mathcal{F}}, \tilde{\mathcal{L}}_1$ such that linear part of $\tilde{\mathcal{L}}_1$ has the form*

$$\begin{pmatrix} I_{v\times v} & S_{v\times o} \\ 0_{o\times v} & I_{o\times o} \end{pmatrix}, \tag{2}$$

where $I_{v\times v}$ is $v \times v$ unit matrix, $I_{o\times o}$ is $o \times o$ unit matrix, $S_{v\times o}$ is a $v \times o$ matrix.

Proof. see [17], Theorem 3.2.

We will use this equivalent key and circulant matrix to reduce the public key size of UOV.

4 Hufu-UOV Signature Scheme

In this section, we reduce the public key size of UOV scheme by using circulant matrix and Toeplitz matrix.

Definition 2. *A matrix of the form*

$$\begin{pmatrix} a_0 & b_1 & b_2 & \cdots & \cdots & b_{n-1} \\ a_1 & a_0 & b_1 & \ddots & \ddots & \vdots \\ a_2 & a_1 & a_0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & b_1 & b_2 \\ \vdots & & \ddots & a_1 & a_0 & b_1 \\ a_{n-1} & \cdots & \cdots & a_2 & a_1 & a_0 \end{pmatrix}$$

is called Toeplitz matrix.

Note that the set of $n \times n$ Toeplitz matrices is a vector space under matrix addition and scalar multiplication. A $n \times n$ Toeplitz matrix is determined by $2n-1$ elements of the first row and column.

Definition 3. *A matrix of the form*

$$\begin{pmatrix} c_0 & c_1 & \cdots & c_{n-2} & c_{n-1} \\ c_{n-1} & c_0 & \cdots & c_{n-3} & c_{n-2} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ c_2 & c_3 & \cdots & c_0 & c_1 \\ c_1 & c_2 & \cdots & c_{n-1} & c_0 \end{pmatrix}$$

is called circulant matrix.

Note that a circulant matrix is determined by the entries of the first row, the other rows are obtained by shifting the previous rows, therefore, we only need to store the first row. Moreover, the multiplication of two circulant matrixes is circulant matrix. The linear combination of circulant matrixes is circulant matrix. The Transpose of circulant matrix is circulant matrix. A $n \times n$ circulant matrix is determined by n elements of the first row.

For simplicity, we set $m = o, v = 3o, n = o + v = 4o$. Let the linear transformation $\mathcal{L}_1$ in Hufu-UOV scheme be

$$\mathcal{L}_1 = L_1\mathbf{x} : \mathbb{F}_q^n \to \mathbb{F}_q^n, \tag{3}$$

where $L_1 \in \mathbb{F}_q^{n \times n}$ is an invertible matrix defined as follow:

$$L_1 = \begin{pmatrix} I_{v\times v} & S \\ 0 & I_{o\times o} \end{pmatrix},$$

where $S \in \mathbb{F}_q^{v\times o}$ is the first o column of $v \times v$ circulant matrix, $I_{v\times v}$ is $v \times v$ identity matrix and $I_{o\times o}$ is $o \times o$ identity matrix.

Let the central map of Hufu-UOV scheme be

$$\mathcal{F} = \{f^{(1)}, f^{(2)}, \cdots, f^{(m)}\} : \mathbb{F}_q^n \to \mathbb{F}_q^m, \tag{4}$$

where

$$f^{(k)}(x_1, \cdots, x_n) = \mathbf{x}^T Q^{(k)} \mathbf{x},$$

and

$$Q^{(k)} = \begin{pmatrix} Q_1^{(k)} & Q_2^{(k)} \\ Q_3^{(k)} & \lambda^{(k)} A \end{pmatrix},$$

where $Q_1^{(k)}$ are $v \times v$ circulant matrixes, $Q_2^{(k)}$ are the first o columns of some $v \times v$ circulant matrixes, $Q_3^{(k)}$ are the first o rows of some $v \times v$ circulant matrixes. A is a $o \times o$ Toeplitz matrix and $\lambda^{(k)} \in \mathbb{F}_q$.

Let $\mathcal{L}_2 = L_2\mathbf{x} : \mathbb{F}_q^m \to \mathbb{F}_q^m$ be an randomly choosen invertible transformation, where $L_2 \in \mathbb{F}_q^{m\times m}$ is an invertible matrix, then the public key map of Hufu-UOV is generated as follow:

$$\mathcal{P} = \mathcal{L}_2 \circ \mathcal{F} \circ \mathcal{L}_1.$$

According to the properties of circulant matrix and Toeplitz matrix, the public key polynomials $p^{(k)}(k = 1, 2, \cdots, m)$ of Hufu-UOV can be expressed as follow:

$$p^{(k)}(x_1, \cdots, x_n) = \mathbf{x}^T P^{(k)} \mathbf{x},$$

where $\mathbf{x}^T = (x_1, \cdots, x_n)$ is a row vector,

$$P^{(k)} = \begin{pmatrix} P_1^{(k)} & P_2^{(k)} \\ P_3^{(k)} & P_4^{(k)} \end{pmatrix}$$

where $P_1^{(k)}$ are $v \times v$ circulant matrixes, $P_2^{(k)}$ are the first o columns of some $v \times v$ circulant matrixes, $P_3^{(k)}$ are the first o rows of some $v \times v$ circulant matrixes, $P_4^{(k)}$ are top left $o \times o$ submatrixes of some $v \times v$ circulant matrixes. In fact $P_4^{(k)}$ are Toeplitz matrixes.

The key generation, signature generation and verification algorithms of Hufu-UOV signature scheme are described in Algorithms 1, 2, 3, 4.

Algorithm 1 `Hufu-UOVKeyGen`: Key Generation of Hufu-UOV

Input:
The Hufu-UOV parameters (q, o, v)
Output:
Hufu-UOV key pairs (sk, pk)
1: Randomly generate invertible transformation $\mathcal{L}_1 : \mathbb{F}_q^n \to \mathbb{F}_q^n$ as (3).
2: Randomly generate invertible transformation $\mathcal{L}_2 = L_2\mathbf{x} : \mathbb{F}_q^m \to \mathbb{F}_q^m$.
3: Randomly generate central map $\mathcal{F} : \mathbb{F}_q^n \to \mathbb{F}_q^m$ as (4).
4: Compute $\mathcal{P} = \mathcal{L}_2 \circ \mathcal{F} \circ \mathcal{L}_1$.
5: $sk = (\mathcal{L}_1, \mathcal{F}, \mathcal{L}_2)$.
6: $pk = \mathcal{P}$.
7: **return** (sk, pk);

Algorithm 2 `HUFU-UOVSmallKeyGen`: Key Generation of HUFU-UOV

Input:
The Hufu-UOV parameters (q, o, v).
Output:
HUFU-UOV key pairs (sk, pk)
1: $(\mathbf{s}_1, \mathbf{s}_2) \leftarrow$ CryptoRandomBytes$(|\mathbf{s}_1|, |\mathbf{s}_2|)$, where $|\mathbf{s}_1|$ and $|\mathbf{s}_2|$ are the bit sizes of $\mathbf{s}_1$ and $\mathbf{s}_2$ respectively;
2: Randomly generate matrix triples $\left(\langle P_1^{(1)}, P_2^{(1)}, P_3^{(1)}\rangle, \cdots, \langle P_1^{(o)}, P_2^{(o)}, P_3^{(o)}\rangle\right)$ by using seed $\mathbf{s}_1$, where $P_1^{(k)}$ is a $v \times v$ circulant matrixes, $P_2^{(k)}$ is a $v \times o$ matrixes which is the first o columns of some $v \times v$ circulant matrixes, and

$P_3^{(k)}$ is a $o \times v$ matrixes which is the first o rows of some $v \times v$ circulant matrixes, $(k = 1, 2, \cdots, o)$.

3: $S \leftarrow$ a $v \times o$ matrix which is the first o columns of a $v \times v$ circulant matrix and generated by using seed $\mathbf{s}_2$.
4: Randomly generate a $o \times o$ Toeplitz matrix A by using seed $\mathbf{s}_2$.
5: Randomly generate $\lambda^{(k)} \in \mathbb{F}_q (k = 1, \cdots, o)$ by using seed $\mathbf{s}_2$.
6: Randomly generate $o \times o$ invertible matrix L_2 by using seed $\mathbf{s}_2$. .
7: Compute matrix triples

$$\left(\langle G_1^{(1)}, G_2^{(1)}, G_3^{(1)}\rangle, \cdots, \langle G_1^{(o)}, G_2^{(o)}, G_3^{(o)}\rangle\right) \leftarrow L_2^{-1}\left(\langle P_1^{(1)}, P_2^{(1)}, P_3^{(1)}\rangle, \cdots, \langle P_1^{(o)}, P_2^{(o)}, P_3^{(o)}\rangle\right).$$

8: $Q_1^{(k)} \leftarrow G_1^{(k)}, (k = 1, \cdots, o)$.
9: $Q_2^{(k)} \leftarrow G_2^{(k)} - G_1^{(k)} S, (k = 1, \cdots, o)$.
10: $Q_3^{(k)} \leftarrow G_3^{(k)} - S^T G_1^{(k)}, (k = 1, \cdots, o)$.
11: $G_4^{(k)} \leftarrow S^T Q_1^{(k)} S + Q_3^{(k)} S + S^T Q_2^{(k)} + \lambda^{(k)} A, (k = 1, \cdots, o)$.
12: $(P_4^{(1)}, \cdots, P_4^{(o)}) \leftarrow L_2\left(G_4^{(1)}, \cdots, G_4^{(o)}\right)$.
13: $sk \leftarrow (\mathbf{s}_1, \mathbf{s}_2)$ or $Q_1^{(k)}, Q_2^{(k)}, Q_3^{(k)}, \lambda^{(k)}, A, S, L_2, (k = 1, \cdots, o)$.
14: $pk \leftarrow \mathbf{s}_1, P_4^{(k)}, (k = 1, \cdots, o)$.
15: **return** (sk, pk);

Algorithm 2 return a key pair of Hufu-UOV signature scheme. The function CryptoRandomBytes() returns a randomly string which is cryptography secure. If we want to get a small private key, we can only store seeds $\mathbf{s}_1, \mathbf{s}_2$ and generate central map and linear transformation by using $\mathbf{s}_1, \mathbf{s}_2$ in process of signature generation. If we want to speed up signature generation, we will store central map and the inverse linear transformations. Since $P_4^{(k)} (k = 1, 2, \cdots, o)$ are Toeplitz matrixes. Therefore the public key size of HUFU-UOV is

$$|pk| = 2o^2 - o + |\mathbf{s}_1|$$

field elements. Comparing to original UOV scheme, we reduce the public key size by a factor which is approximately equal to $\lfloor 4.5o \rfloor$.

Algorithm 3 `Hufu-UOVSign`: Signature Generation of Hufu-UOV

Input:
Document $\mathbf{m}$ and sk
Output:
Signature $(s_1, \cdots, s_n)||\mathbf{r}$

1: $\mathbf{r} \leftarrow \ell$ bits string which is generated randomly.
2: $(h_1, \cdots, h_m) \leftarrow \mathcal{H}(\mathcal{H}(\mathbf{m})||\mathbf{r})$.
3: Compute $(y_1, \cdots, y_m) = \mathcal{L}_2^{-1}(h_1, \cdots, h_m)$
4: $eof \leftarrow 0$.
5: **while** $eof == 0$ **do**
6: Randomly choose$(\bar{x}_1, \cdots, \bar{x}_v) \in \mathbb{F}_q^v$.

Substitute these values into $\mathcal{F}(\bar{x}_1, \cdots, \bar{x}_v, x_{v+1}, \cdots, x_n) = (y_1, \cdots, y_m)$. We can obtain a linear system and a multivariate quadratic equation with unknowns $x_{v+1}, \cdots, x_n$.
Solve the linear system and obtain a solution $\bar{x}_{v+1} = a_1 x_n, \bar{x}_{v+2} = a_2 x_n, \cdots, \bar{x}_{n-1} = a_{o-1} x_n$.
Substitute $(\bar{x}_1, \cdots, \bar{x}_{n-1}, x_n)$ into the multivariate quadratic equation. We can obtain a univariate quadratic equation with unknown x_n.
if The univariate quadratic equation has solution **then**
    Solve this univariate quadratic equation and get a solution $x_n = \bar{x}_n$. Finally, we get a solution $(\bar{x}_1, \cdots, \bar{x}_n)$ such that $\mathcal{F}(\bar{x}_1, \cdots, \bar{x}_n) = (y_1, \cdots, y_m)$.
    $eof \leftarrow 1$.
  end if
end while
$(s_1, \cdots, s_n) \leftarrow \mathcal{L}_1^{-1}(\bar{x}_1, \cdots, \bar{x}_n)$.
return $(s_1, \cdots, s_n)||\mathbf{r}$;

Algorithm 3 generates a signature for a given document. The signature includes ℓ bits salt $\mathbf{r}$.

Algorithm 4 `Hufu-UOVVer`: Verification of Hufu-UOV

Input:
Signature $(s_1, \cdots, s_n)||\mathbf{r}$, document $\mathbf{m}$ and public key pk
Output:
True if $(s_1, \cdots, s_n)$ is a valid signature for $\mathbf{m}$, **False** otherwise
$(h_1, \cdots, h_m) \leftarrow \mathcal{H}(\mathcal{H}(\mathbf{m})|\mathbf{r})$.
$(h'_1, \cdots, h'_m) \leftarrow \mathcal{P}(s_1, s_2, \cdots, s_n)$.
if $(h'_1, \cdots, h'_m) == (h_1, \cdots, h_m)$ **then**
  return True
else
  return False
end if

Algorithm 4 verify whether the signature $(s_1, \cdots, s_n)||\mathbf{r}$ is indeed a valid signature.

5 Security of Hufu-UOV

The security of UOV and Rainbow signature schemes have been well studied. The methods which can be used to attack Hufu-UOV scheme are showed in table 1.

5.1 Direct Attack

Given a document $\mathbf{y} = (y_1, \cdots, y_m) \in \mathbb{F}_q^m$, a straightforward method to attack Hufu-UOV scheme is to try to solve the public system $\mathcal{P}(\mathbf{x}) = \mathbf{y}$. If we find a

Table 1. Methods of attacking Hufu-UOV.

Attack methods	Complexity	
	Classic	Quantum
Direct attack[12,13]	$\mathcal{O}\left(q^k\binom{n-k+d_{reg}(k)}{d_{reg}(k)}^{\omega}\right)$	$\mathcal{O}\left(q^{k/2}\binom{n-k+d_{reg}(k)}{d_{reg}(k)}^{\omega}\right)$
UOV attack [2]	$\mathcal{O}\left(q^{v-o}n^4\right)$	$\mathcal{O}\left(q^{\frac{v-o}{2}}n^4\right)$

vector $\bar{\mathbf{x}} = (\bar{x}_1, \cdots, \bar{x}_n) \in \mathbb{F}_q^n$ satisfies $\mathcal{P}(\bar{\mathbf{x}}) = \mathbf{y}$, then $\bar{\mathbf{x}}$ is the forgery signature of $\mathbf{y}$. This is so called direct attack. To achieve this, the attacker can use a Gröbner Basis method such as F4 or F5 algorithm [12,13]. Since the public key system of Hufu-UOV scheme is underdetermined ($m < n$), one usually fixes some of the variables before applying F4 or F5 algorithm. In [15], Barget et al. determined the computation complexity for F4 and F5 algorithms over the finite field $\mathbb{F}_q$ to be

$$\mathcal{O}\left(q^k\binom{n-k+d_{reg}(k)}{d_{reg}(k)}^{\omega}\right)$$

where $\omega = 3$ in the usual Gaussian elimination algorithm and $\omega = 2.3766$ in improved algorithm, $d_{reg}(k)$ is the degree of regularity of the ideal formed by the polynomials in the system after fixing k variables; it is given by the degree of the first term with negative coefficient in the expansion of

$$\frac{\prod_{i=1}^{m}(1-z^{d_i})}{(1-z)^n},$$

with d_i being the total degree of the i-th equations.

Since Grovers algorithm can be used to speed up the brute force part of the direct attack by using quantum computer, thought there are no specialized quantum algorithms have been found to solve polynomial system over finite fields, thus the new complexity of direct attack by using quantum computer is

$$\mathcal{O}\left(q^{k/2}\binom{n-k+d_{reg}(k)}{d_{reg}(k)}^{\omega}\right).$$

5.2 UOV attack

The UOV attack was proposed by Kipnis and Shamir [8]. The goal of this attack is to find the pre-image of the so called Oil subspace. Finding this space allows to separate the oil from the vinegar variables and recovering the private key. the complexity of UOV attack is showed in table 1.

6 Parameters

Based on the security analysis in the previous section, we propose some parameters for Hufu-UOV signature scheme.

6.1 Parameters of Hufu-UOV

We choose the parameters $m = o, v = 3o, n = o + v = 4o$ for Hufu-UOV scheme as the following table 2 shows. The size of signature include 16 bytes salt, that is $\ell = 128$.

Table 2. Parameters of HUFU-UOV signature scheme.

Parameters $(\mathbb{F}_q, o, v)$	security level(bits)	public key size (KB)	private key size (Bytes)	hash value size (Bytes)	signature size (Bytes)
$(\mathbb{F}_{16}, 64, 128)$	128	4.096	64	32	112
$(\mathbb{F}_{16}, 96, 192)$	192	9.216	96	48	144
$(\mathbb{F}_{16}, 128, 256)$	256	16.384	128	64	192
$(\mathbb{F}_{256}, 48, 96)$	128	4.592	64	48	160
$(\mathbb{F}_{256}, 72, 144)$	192	10.344	96	72	232
$(\mathbb{F}_{256}, 96, 192)$	256	18.4	128	96	304

6.2 Comparison of UOV-based Signature Schemes

Table 3 compares the key and signature sizes of UOV-based signature schemes on 128 bits security level. Note that Hufu-UOV signature scheme includes 128 bits salt in signatures.

Table 3. Comparison the key and signature sizes of UOV-based signature schemes.

scheme	public key size (KB)	hash size (Bytes)	signature size (Bytes)	quantum resistant
UOV	508.08	48	144	yes
Rainbow[3]	187.7	48	88	yes
UOVrand[17]	52.531	45	135	yes
RainbowLRS2[17]	44.5	43	79	yes
Hufu-UOV	**4.096**	**32**	**112**	**yes**

7 Conclusion

In this paper, we propose a new method to reduce the public key size of UOV signature scheme. This method can make UOV more practical. For example. we can use Hufu-UOV to build a quantum resistant block chain.

References

1. M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman, 1979.
2. Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced Oil and Vinegar Signature Schemes. In EUROCRYPT, volume 1592 of Lecture Notes in Computer Science, pages 206-222. Springer, 1999.
3. Jintai Ding and Dieter Schmidt. Rainbow, a New Multivariable Polynomial Signature Scheme. In ACNS, volume 3531 of Lecture Notes in Computer Science, pages 164-175, 2005.
4. Jacques Patarin, Nicolas Courtois, and Louis Goubin. QUARTZ, 128-Bit Long Digital Signatures. In CT-RSA, volume 2020 of Lecture Notes in Computer Science, pages 282-297. Springer, 2001.
5. Ward Beullens, Bart Preneel. LUOV: Signature Scheme proposal for NIST PQC Project. 2017. Technical Report. National Institute of Standards and Technology. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.
6. Aviad Kipnis and Adi Shamir. Cryptanalysis of the HFE public key cryp- tosystem. In Advances in Cryptology - CRYPTO 1999, volume 1666 of Lecture Notes in Computer Science, pages 19-30. Michael Wiener, ed- itor, Springer, 1999.
7. Patarin J.: The oil and vinegar signature scheme. Presented at the Dagstuhl Work- shop on Cryptography (1997).
8. Aviad Kipnis and Adi Shamir. Cryptanalysis of the Oil and Vinegar Signature Scheme[A]. In CRYPTO, volume 1462 of Lecture Notes in Computer Science[C], Heidelberg: Springer,1998: 257-266
9. Olivier Billet and Henri Gilbert. Cryptanalysis of Rainbow[A]. In SCN, volume 4116 of Lecture Notes in Computer Science[C], Berlin Heidelberg: Springer, 2006: 336-347
10. Louis Goubin and Nicolas Courtois. Cryptanalysis of the TTM Cryptosystem. In ASIACRYPT, volume 1976 of Lecture Notes in Computer Science, pages 44-57. Springer, 2000.
11. D. Coppersmith, J. Stern, S. Vaudenay: Attacks on the birational signature scheme. CRYPTO 1994, LNCS vol. 773, pp. 435 - 443. Springer, 1994.
12. Jean-Charles Faugère. A New Efficient Algorithm for Computing Gröbner Bases (F4). Journal of Pure and Applied Algebra, 139(1U3):61-88, 1999.
13. Jean-Charles Faugère. A New Efficient Algorithm for Computing Gröbner Bases without Reduction to Zero (F5). In Proceedings of the 2002 International Symposium on Symbolic and Algebraic Computation, ISSAC, pages 75-83. ACM, 2002.
14. Enrico Thomae and Christopher Wolf. Solving underdetermined systems of multivariate quadratic equations revisited. In International Workshop on Public Key Cryptography, pages 156C171. Springer, 2012.
15. Bardet, M., Faugère, J.-C., Salvy, B.: On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations. In: International Conference on Polynomial System Solving - ICPSS, pages 71-75. Nov 2004
16. Ward Beullens, Bart Preneel. LUOV: Signature Scheme proposal for NIST PQC Project. 2017. Technical Report. National Institute of Standards and Technology. available at https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions.

17. Albrecht Petzoldt. Selecting and reducing key sizes for multivariate cryptography. 2013.
18. A. Petzoldt, S. Bulygin and J. Buchmann, "A multivariate signature scheme with a partially cyclic public key," in Proc. of Proceedings of SCC 2010. Citeseer, 2010.
19. A. Petzoldt and S. Bulygin, "Linear recurring sequences for the UOV key generation revisited," Information Security and CryptologyCICISC 2012. Springer, pp. 441C455, 2013.
20. Wolf C, Preneel B. Equivalent keys in Multivariate Quadratic public key systems[J]. Journal of Mathematical Cryptology, 2011, 4(4): 375-415.
21. D. Coppersmith, J. Stern, S. Vaudenay: Attacks on the birational signature scheme. CRYPTO 1994, LNCS vol. 773, pp. 435-443. Springer, 1994.
22. J. Ding, B.-Y. Yang, C.-H. O. Chen, M.-S. Che, C.-M. Cheng: New differential-algebraic attacks and reparametrization of Rainbow. ACNS 2008, LNCS vol. 5037, pp. 242-257. Springer, 2008.
23. Jintai Ding, Joshua Deaton, Vishakha, Bo-Yin Yang: The Nested Subset Differential Attack - A Practical Direct Attack Against LUOV Which Forges a Signature Within 210 Minutes. EUROCRYPT (1) 2021: 329-347