Circulant UOV: a new UOV variant with shorter private key and faster signature generation

Zhiniang Peng and Shaohua Tang*
School of Computer Science & Engineering, South China University of Technology
Guangzhou, China
[e-mail: 246003@qq.com csshtang@scut.edu.cn]
*Corresponding author: Shaohua Tang

Received April 2, 2017; revised June 1, 2017; accepted October 11, 2017; published March 31, 2018

Abstract

UOV is one of the most important signature schemes in Multivariate Public Key Cryptography (MPKC). It has a strong security guarantee and is considered to be quantum-resistant. However, it suffers from large key size and its signing procedure is relatively slow. In this paper, we propose a new secure UOV variant (Circulant UOV) with shorter private key and higher signing efficiency. We estimate that the private key size of Circulant UOV is smaller by about 45% than that of the regular UOV and its signing speed is more than 14 times faster than that of the regular UOV. We also give a practical implementation on modern x64 CPU, which shows that Circulant UOV is comparable to many other signature schemes.

Keywords: MPKC; UOV Signature Scheme; Post-Quantum Cryptosystem; AVX2

This work was supported by the National Natural Science Foundation of China (Nos. 61632013, U1135004 and 61170080), 973 Program (No. 2014CB360501), Guangdong Provincial Natural Science Foundation (No. 2014A030308006), and Guangdong Provincial Project of Science and Technology (no. 2016B090920081).

http://doi.org/10.3837/tiis.2018.03.022 **ISSN : 1976-7277**

1. Introduction

In [1] [2], Shor proposed some polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. It posed a serious threat to some existing cryptographic schemes such as RSA and ECC, which are based on those problems. After that, Post-Quantum Cryptography [3] [4], which is secure against attacks by a quantum computer, became a very important research area. MPKC (Multivariate Public Key Cryptography) is one of the most promising candidates for Post-Quantum Cryptography.

Since the first MPKC scheme: MI [5] was proposed in 1985, this area has undergone a rapid development in last two or three decades. A lot of MPKC encryption and signature schemes have been proposed, e.g., TTS [6], MQQ [7], ABC [8], HFE [9], ZHFE [10], EFC [50] etc. However, most of them were broken by various attacks, such as MinRank [6] , HighRank attack [11] [6], Direct attack, Differential attack [46] and Rainbow-Band-Separation attack [12] [11]. Despite this, UOV (Unbalanced Oil and Vinegar scheme) remains secure for almost two decades. None of the existing attacks can cause severe security threats to it.

However, like other MPKC schemes, UOV has not been widely used. One reason is that it has a large key size. The other reason is that UOV is not known as the fastest multivariate signature scheme. Many MPKC signature schemes such as Gui [13] and Rainbow [14] outperform UOV in signing speed. To make UOV practical, we have to reduce the key size and improve the signing speed. Some work has been done to reduce the public key size of UOV or UOV-like schemes. In [15] [16] [17], the authors proposed Cyclic UOV to insert some special sequences into the generation of public key to reduce the public key size. The public key size is reduced by 83% compared with regular UOV. In addition, the verification procedure is sped up according to the conclusion made in [18].

Several variants of Rainbow or UOV using sparse private keys have been proposed to reduce the private key size. TTS and enhanced TTS are such schemes. The parameters proposed in the original paper of TTS and enhanced TTS are broken, but the method of reducing the size of the private key which use the sparse key is surviving now. In general, these schemes use three methods, including PRNG (Pseudorandom Number Generator) method, Matrix-based method and NT method, to construct their central maps.

PRNG method is to reduce the private key size by using a pseudorandom number generator [19] [20]. This method can reduce the key size to a constant, but it will also decrease the signature generation performance seriously. Matrix-based method divides the central maps of Rainbow into smaller blocks by using diagonal matrix representations [21] [22]. This method reduces the private key size of Rainbow and requires one to solve a much smaller systems of linear equations. The NT method introduce some rotating relations into Vinegar-Vinegar terms of the central maps of Rainbow [23] so that the same matrix computation appears several times.

Matrix-based method and NT method were originally designed for Rainbow. As UOV can be considered as a single layer Rainbow, these two methods can also be applied to UOV. In [24], authors proposed MB-UOV by combining these methods. However, we find out that associated symmetric matrices P'_i of public key polynomials P_i of Matrix-based UOV leak the a subspace $T^{-1}(O)$ with dimension o-d. Attackers can form an equivalent key of MB-UOV using this subspace in polynomial time.

Our Contributions: In this paper, we propose Circulant UOV with higher signature generating efficiency and shorter private key. Our contribution is twofold. On the theoretical side, we explore a new method to construct a UOV signature scheme with higher signature generating efficiency and shorter private key. Analysis and theories are provided to show the security and efficiencies of our proposed scheme. On the practical side, we implement our Circulant UOV and other UOV variants on a Intel Core i7-4790 @3.60Ghz CPU. Experiments shows that Circulant UOV is better than all the existing UOV variants in terms of signature generation and private keysize. We also give an overall comparison with Gui, Rainbow, GLP, RSA and ECDSA under different security requirements. The results show that Circulant UOV outperforms many other signature schemes in both signing and verification speed.

2. UOV

In this section, we will introduce UOV and its variants. For the convenience of readers, the meaning of notations can be found in the table of notations in appendix.

2.1 Basic UOV

UOV is a modified version of the Oil and Vinegar scheme designed by J. Patarin [26] to prevent OV attack [25]. It poses a strong security and none of the existing attacks can cause severe security threats to it.

To figure out what UOV is, first of all, we'd like to introduce the concept of Oil-Vinegar polynomial with the following form:

$$f=\sum_{i=1}^{v}\sum_{j=1}^{v}a_{ij}x'_i\,x'_j+\sum_{i=1}^{o}\sum_{j=1}^{v}b_{ij}\overline{x_i}x'_j+\sum_{j=1}^{v}\beta_j x'_j+\sum_{i=1}^{o}\alpha_i\overline{x_i}+c\,.$$

Variables are divided into two kinds in the above polynomial: Oil variables $\overline{x_i}$ and Vinegar variables x'_j. The number of Oil variables is o and the number of the Vinegar variables is v. Central map F can be composed of o Oil-Vinegar polynomials over the base field K. The invertibility of the central map comes from the fact that once random values are assigned to the Vinegar variables set, it becomes a set of linear equations of Oil variables and can be solved by Gauss Elimination.

Once the central map F: $K^n \to K^m$ is determined, the public key P can be calculated as: $P=F\circ T : K^n \to K^m$, in which T: $K^n \to K^n$ is an affine transformation. The inverse of P can be computed as follows:

Step 1 Randomly choose $v_1,\ldots, v_v \in K$.

Step 2 Substitute $(x_1,\ldots,x_v)$ with $(v_1,\ldots,v_v)$, we will get o linear equations of o variables. Solve the system and obtain a solution $\overline{x_1},\ldots,\overline{x_o}$ (If the system is not regular, go back to Step 1). Let $(x_1,\ldots,x_n)=(x'_1,\ldots,x'_v,\overline{x_1},\ldots,\overline{x_o})$.

Step 3 Apply inverse map of T to $(x_1,\ldots,x_n)$.

Define $d=v-o$. When $d=0$, it's called balanced Oil-Vinegar scheme (OV for short). When $d>0$, it's known as UOV [26].

2.2 UOV Variants

In [15] [16] [17], the authors proposed Cyclic UOV to insert some special sequences into the generation of public key to save some memory. Many people follow this work because the

cyclic method reduces UOV public key size and improves verification speed. It also enjoys strong security guarantee just like regular UOV.

Cyclic UOV can be used to reduce public key size and improve verification speed of UOV. It is not difficult to reduce the private key size of UOV in comparison with the reduction of public key size. For example, one can use a pseudorandom number generator to compress the private key size of UOV into a constant number. But this method will severely reduce the signing speed.

Matrix-based Rainbow and NT Rainbow, which use sparse private keys, have been proposed to reduce the private key size and improve the signing speed. As UOV can be considered as a single layer Rainbow, these techniques can also be applied to UOV. In [24], Tan et al. proposed MB-UOV with smaller private key and faster signature generation by combining these methods.

We find out that associated symmetric matrices P'_i of public key polynomials P_i of Matrix-based UOV leak a subspace $T^{-1}(O)$ with dimension o-d, where d is usually equal to 2 or 3 in Matrix-based UOV. This is because the quadratic matrices F'_i of the central map polynomials F_i of Matrix-based UOV is not full rank and each kernel of them happens to be a subspace of Oil Space O. Attackers can form an equivalent key of Matrix-based UOV using this subspace in polynomial time[1]. This is actually a kind of Rank attack. One may think we can apply a left affine transformation S into Matrix-based UOV public key to block this attack. But this will not solve the problem fundamentally because attackers can use the HighRank attack to eliminate the impact of S. In order to increase the complexity of HighRank attack, we should choose large basic field for Matrix-based UOV. This will severely degrade the performance of Matrix-based UOV. It should be noted that authors have already considered this kind of attack in Matrix-based Rainbow [22].

Although Matrix-based method is not suitable for UOV, The NT method is a good way to reduce the private key size and improve the signing speed of UOV.

3. A NEW SECURE UOV VARIANT

In this section, we propose a new variant of UOV, called Circulant UOV. Although its name is similar to Cyclic UOV, the basic ideas are quite different. In Cyclic UOV, the authors manage to reduce the public key size and verification complexity. But in our Circulant UOV, we aim at reducing the private key size and improving signing speed. We will begin by explaining the basic idea underlying our scheme.

3.1 Basic Underlying Idea

The basic idea underlying our scheme is to speed up Step 2 of UOV signing process, which is the slowest part of the signing algorithm. In Step 2 of UOV signature generation process, we need to solve a system of linear equations described as $LX=V$, where L is a o-by-o square matrix and V is a column vector of size o, X is a size o column vector of variables. In general, we use Gauss Elimination to find X, which is very costly. In Circulant UOV, we put some rotating relations in parts of UOV central matrices to make L become a circulant matrix. Here we define an o-by-o circulant matrix L taking the following form:

[1] Code of our attack can be found at https://github.com/edwardz246003/MB-UOV-attack.

$$L=\begin{pmatrix} l_1 & l_2 & \cdots & l_{o-1} & l_o \\ l_o & l_1 & \cdots & l_{o-2} & l_{o-1} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ l_3 & l_4 & \cdots & l_1 & l_2 \\ l_2 & l_3 & \cdots & l_o & l_1 \end{pmatrix}.$$

The inverse of a circulant matrix can be computed very efficiently by using Extended Euclidean algorithm. In addition, the structure introduced in private key will also improve the speed of the remaining parts of Step 2 significantly.

3.2 Private Key of Circulant UOV

At first, we show that the matrix representation of our new UOV central polynomials. We keep the constant and linear parts so that our central matrices are *(n+1)*-by-*(n+1)* matrices of the form in **Fig. 1**.

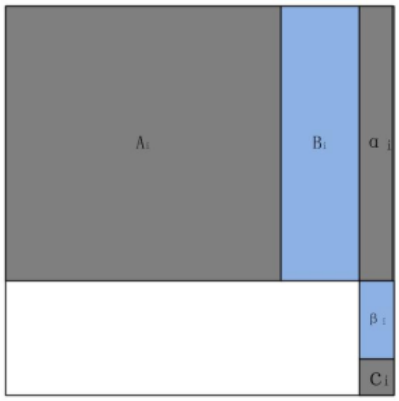

Fig. 1. Central matrix of Circulant UOV.

The white area stands for zero elements. The gray areas mean arbitrary elements in the base field. The blue areas will have some rotating relations with other central matrices. Submatrix A_i is a *v*-by-*v* matrix standing for Vinegar-Vinegar cross-terms coefficients, B_i is a *v*-by-*o* matrix standing for Oil-Vinegar cross-terms. $\boldsymbol{\beta}_i$ in the last column is the linear coefficients of Oil variables. $\boldsymbol{\alpha}_i$ is the linear coefficients of Vinegar variables and c_i is the constant term. Every single central matrix of Circulant UOV looks exactly the same as regular UOV.

In fact, A Circulant UOV doesn't have circulant matrix in its central matrices. It only has some rotating relations among parts of submatrix of different central matrices. Those rotating relations will help us get a circulant matrix during the signing process. If we write matrix B_i in column form, B_i and $\boldsymbol{\beta}_i$ have the following rotating relations:

$$\begin{aligned} B_1&=(\boldsymbol{b}_1,\boldsymbol{b}_2,\ldots,\boldsymbol{b}_o) & \boldsymbol{\beta}_i&=(\beta_1', \beta_2',\ldots, \beta_o') \\ B_2&=(\boldsymbol{b}_o,\boldsymbol{b}_1,\ldots,\boldsymbol{b}_{o-1}) & \boldsymbol{\beta}_i&=(\beta_o', \beta_1',\ldots, \beta_{o-1}') \\ &\vdots & &\vdots \\ B_o&=(\boldsymbol{b}_2,\boldsymbol{b}_3,\ldots,\boldsymbol{b}_1) & \boldsymbol{\beta}_i&=(\beta_2', \beta_3',\ldots, \beta_1') \end{aligned}$$

3.2 Inverting the Central Map

In this section, we are going to describe how to invert the central map of Circulant UOV.

Assume the value to be inverted is M. We randomly choose a Vinegar vector $\boldsymbol{v}$. Substituting $(x_1,\ldots,x_v)$ with $(v_1,\ldots,v_v)$, we will get a linear equation system of o variables. For each central polynomial P_k we get an equation:

$$\overbrace{\mathbf{v}^T \cdot \mathbf{A_k} \cdot \mathbf{v} + \mathbf{v}^T \cdot \mathbf{a_k} + c_k}^{constant} + \overbrace{\mathbf{v}^T \cdot \mathbf{B_k} \cdot \mathbf{o} + \boldsymbol{\beta}_\mathbf{k} \cdot \mathbf{o}}^{linear\ in\ \mathbf{o}} = y_k ,$$

where vector $\boldsymbol{o}=(o_1,\ldots,o_o)$ stands for Oil variables. Let $u_k = y_k - (\mathbf{v}^T \cdot \mathbf{A_k} \cdot \mathbf{v} + \mathbf{v}^T \cdot \mathbf{a_k} + c_k)$ for $k \in [1,\ldots,o]$. Then we get a linear system:

$$\underbrace{\begin{pmatrix} \mathbf{v}^T \cdot \mathbf{B_1} + \boldsymbol{\beta}_\mathbf{1} \\ \mathbf{v}^T \cdot \mathbf{B_2} + \boldsymbol{\beta}_\mathbf{2} \\ \vdots \\ \mathbf{v}^T \cdot \mathbf{B}_{\mathbf{o\text{-}1}} + \boldsymbol{\beta}_{o\text{-}\mathbf{1}} \\ \mathbf{v}^T \cdot \mathbf{B}_o + \boldsymbol{\beta}_o \end{pmatrix}}_{\mathbf{L}} \cdot \begin{pmatrix} o_1 \\ o_2 \\ \vdots \\ o_{o-1} \\ o_o \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \\ \vdots \\ u_{o-1} \\ u_o \end{pmatrix}.$$

Matrix B_i and vector $\boldsymbol{\beta}_i$ have rotating relations. After plugging in $\boldsymbol{v}$, L will be an o-by-o circulant matrix, which can be inverted efficiently.

Computing $\boldsymbol{L}$: Before talking about how to invert L, we first introduce how to calculate L efficiently. To compute matrix L, we need to compute $\boldsymbol{v}^T * B_k + \boldsymbol{\beta}_k$ for $k \in [1,\ldots,o]$. Since B_k is generated by cyclically right rotating k rows of B_1 and $\boldsymbol{\beta}_k$ is generated by cyclically down rotating k elements of $\boldsymbol{\beta}_1$, the signer only needs to compute the first row of L. The rest of L are generated by its right rotating sequences. The time complexity of computing L is improved by a factor of o.

Invertible probability of $\boldsymbol{L}$: If the matrix L we get is not invertible, we have to choose another random Vinegar vector $\boldsymbol{v}$ to get another invertible matrix L. To get a faster signing algorithm, we have to make sure a random L is invertible with high probability. Here we test the invertible probability $P(o)$ of a random o-by-o circulant matrix over GF(31) by experiments. We test each $P(o)$ with $o \in [30,120]$ for 10^5 times and calculate their average values. Results are presented in **Fig. 2**.

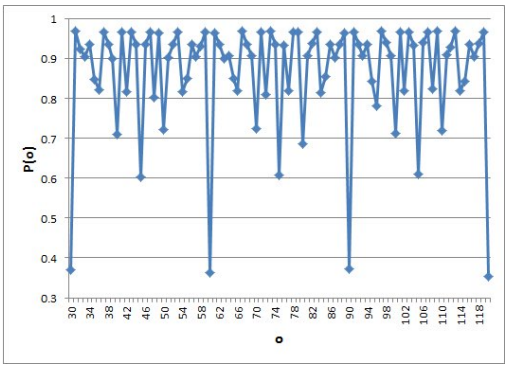

Fig. 2. Estimated value of $P(o)$.

It seems that $P(o)$ is a periodic function. For most of o, $P(o)$ is close to $30/31$, which is the probability of a random matrix over GF(31). But $P(o)$ will be much smaller if o is a multiple of 5 or 6, so we have to avoid them when choosing parameters.

Inverting L: In this paper, we use Extended Euclidean algorithm to compute the inverse of a circulant matrix L. Here we give a simple description of it. Suppose L is an invertible circulant matrix over a finite field Z_p. We consider the problem of computing a circulant matrix J such that $LJ = I$ (It is well known that the inverse of a circulant matrix is still circulant). Let $(l_0,l_1,\ldots,l_{o-1})$ be the first row of L. It is natural to associate L with the polynomial $f(x)=\sum_{i=0}^{o-1} l_i x^i$ (over the ring $Z_p[x]$). Computing the inverse of L is equivalent to finding a polynomial $g(x)$ in $Z_p[x]$ such that $f(x)*g(x)=1 \bmod x^o-1$ [27]. Hence, the problem of inverting a circulant matrix is equivalent to inverting a polynomial in the ring $Z_p[x]/(x^o-1)$. It takes about $O(o^2)$ arithmetic operations by using the Extended Euclidean algorithm. In short, inverting L of Circulant UOV is much faster than that of regular UOV. **Table 1** gives a comparison of them.

Table 1. A comparison between regular UOV and Circulant UOV in inverting L.

	Circulant UOV	Regular UOV
Computing L	$O(o^2)$	$O(o^3)$
Computing L^{-1}	$O(o^2)$	$O(o^3)$

3.4 Size of Central Map

Based on the previous description of how to construct Circulant UOV, we list the parameters needed to build the central map:

1) B_1: the initial v-by-o matrix corresponding to the coefficients of cross-terms between Vinegar variables and Oil variables.
2) $\boldsymbol{\beta}_1$: the initial o dimension vector corresponding to the coefficients of linear terms in Oil variables.
3) A_k for $k \in [1,\ldots,o]$: o random v-by-v matrices.
4) $\boldsymbol{a}_k$ for $k \in [1,\ldots,o]$: all the v dimension vectors corresponding to the coefficients of linear terms in Vinegar variables.
5) c_k for $k \in [1,\ldots,o]$: all constant terms for central polynomials.

We can estimate that the size of Circulant UOV central map is given by

$$o \cdot \left(\frac{v \cdot (v+1)}{2} + v + 1\right) + o \cdot v + o$$

field elements. For reasonable parameters, it's about 45% smaller than and the private key size of regular UOV which is given by

$$o \cdot \left(\frac{v \cdot (v+1)}{2} + o \cdot v + n + 1\right)$$

field elements.

3.5 General Description of Circulant UOV

Here, we give a general description of Circulant UOV. Compared with regular UOV, a left affine transformation S is added into our Circulant UOV to increase security. Affine transformation can be computed efficiently, especially by using AVX2 instructions. So it will not affect the signing speed.

Key generation: According to the required security level, we choose the appropriate set of parameters which include the finite field $K=GF(q)$, number of Vinegar variables v and number of Oil variables o. Set $n=v+o$. Then we generate the key as follows.

1) Randomly choose (B_1, β_1, T, S) and (A_k, a_k, c_k) for $k \in [1,\ldots,o]$.
2) Use (B_1, β_1) to generate (B_k, β_k) for $k \in [2,\ldots,o]$ to construct central map F.
3) Compute $P=S \circ F \circ T$: $K^n \rightarrow K^o$ as public key.
4) Store $(B_1, \beta_1, T^{-1}, S^{-1})$ and (A_k, a_k, c_k) for $k \in [1,\ldots,o]$ as private key.

Signature generation: Suppose the document to be signed is M, then we sign it as follows:

1) Hash it to $Y \in K^o$.
2) Apply affine transformation S^{-1} to Y. Let $Y'=S^{-1}(Y)$.
3) Use method mentioned in Section 3.3 to invert the central map: $X'=F^{-1}(Y')$.
4) Invert the linear affine transformation: $X=T^{-1}(X')$. Output $X \in K^n$ as signature.

Signature verification: The signer sends document-signature pair (M,X) to a receiver. Receiver verifies the correctness of the signature by checking if $P(X)=Hash(M)$. If it matches, the signature is legitimate. Otherwise, reject it.

4. SECURITY OF CIRCULANT UOV

In this section, we are going to analyze the security of Circulant UOV by applying known existing attacks to it.

In Circulant UOV, the public key is a composite mapping $P=S \circ F \circ T$. Since all the components of the central map F are Oil and Vinegar polynomials, components of $S \circ F$ will also be Oil and Vinegar polynomials (These polynomials will no longer have the circulant structure). This means that Circulant UOV is indeed a subset of normal UOV, but we will show that it is as hard as regular UOV if we choose the parameters appropriately.

4.1 Direct Attack

The most straightforward method to attack the UOV signature scheme is to solve the public system $P(X)=Y$ directly. When solving an underdetermined multivariate nonlinear polynomial system, it is often a good strategy to guess some variables to create an overdetermined system. This is called Hybrid approach. We carried out a number of experiments with MAGMA [28], which contains an efficient implementation of F4 algorithm [29] [30] for computing Gröbner bases [31]. We generate 100 random instances for each Circulant UOV and regular UOV and record their average performance against F4 algorithm with Hybrid approach. **Table 2** shows the results of our experiments on attacking Circulant UOV and regular UOV over base fields GF(31) and GF(5).

Table 2. A comparison between regular UOV and Circulant UOV against Direct attack.

Parameters (K,n,m)	Regular UOV	Circulant UOV
(GF(31),9,3)	0.312s	0.305s
(GF(31),12,4)	16.379s	16.511s
(GF(31),15,5)	893.967s	889.658s
(GF(5),9,3)	0.289s	0.295s
(GF(5),12,4)	10.812s	10.774s
(GF(5),15,5)	547.251s	544.93s

From **Table 2**, we can see that the attack time for Circulant UOV is extremely close to that for regular UOV. We can conclude that Circulant UOV and regular UOV almost have the same performance against Direct attack. We can estimate that public systems of Circulant UOV and UOV have the same degree of regularity. As public systems of UOV behave very similar to random systems [32], the degree of regularity dreg of public systems of Circulant UOV can be given as the lowest integer D for which coefficient of z^D in $\frac{(1-z^2)^m}{(1-z)^n}$ is less or equal to 0. So the lower bound of the complexity of Direct attack against Circulant UOV using Hybrid approach can be estimated by [32] [33]

$$HF_5(q,m,n) = \min_{0\le k\le m} q^k \cdot O(m \cdot \binom{n-k+d_{reg}-1}{d_{reg}}^2).$$

4.2 UOV Reconciliation Attack

In UOV scheme, the lower right corner of F_i' must be zero. UOV Reconciliation attack [34] exploits this feature to yield some quadratic equations. Attackers have to solve a system of $(n-j)*o$ quadratic equations in v variables $(j = n-1,\ldots,v)$. If we choose o and v large enough, UOV Reconciliation attack will not harm the security. As Circulant UOV also have UOV key structure, attackers having an equivalent key of $S \circ F$ can forge signatures. So the original UOV Reconciliation attack against Circulant UOV can be applied in the same way as against regular UOV.

One may think that the rotating relations in Circulant UOV might actually give more equations. For example, we have $B_1[1,1]\ B_2[1,2]=0$. As S involved in, attackers will get cubic equations in more variables. He may use the rotating relations to get more equations, but this will not help him to solve the variables. The complexity of this attack is mainly given by the complexity of solving $v*o+o-v$ cubic equations and o quadratic equations in o^2+o-v variables. The complexity of solving such a system using Gröbner basis can be found in [35] [36] [12]. If we choose o and v large enough, it will be infeasible to solve such a system.

4.3 Rainbow-Band-Separation Attack

Rainbow-Band-Separation attack is used to break Rainbow. The intrinsic idea of this attack is to exploit the sparse key structure. Attackers will find an equivalent key (S',F',T') satisfying that $P = S' \circ F' \circ T'$. In Circulant UOV, we have dense cross-terms. In the associated symmetric matrix F'_i of central map, every column looks random. Attackers can not identify zero elements to run Rainbow-Band-Separation attack.

Some people may think that the rotating relations might give some equations in this attack. This is incorrect because the equivalent key S' and T' in Rainbow-Band-Separation have specific forms, thus F' doesn't have rotating relations anymore. So Rainbow-Band-Separation attack is not applicable to Circulant UOV.

4.4 UOV Attack

The goal of UOV attack is to find the preimage of Oil subspace O under transformation T^{-1} by exploiting a symmety hidden in the differential structure of UOV. It can be considered a Differential attack [47]. Attackers form random linear combinations of public quadratic matrices $W = \sum_0^{o-1} \lambda_i P_i'$, multiple them with the inverse of one P_j'. The complexity of the UOV

attack can be estimated by $q^{v-o-1}*o^4$ for UOV. Circulant UOV and UOV have the same preimage of Oil subspace O under transformation T^{-1}. The probability of $W \cdot P_j$' having a non trivial invariant subspace which is also a subspace of $T^{-1}(O)$ is about q^{o-v}. We generate 100 instances of both regular UOV and Circulant UOV and attack them by UOV attack. **Table 3** shows that Circulant UOV and regular UOV almost have the same performance against UOV attack.

Table 3. The time required to find a basis of $T^{-1}(O)$ in UOV attack.

(GF(31),*o*,*v*)	(8,12)	(10,14)	(8,14)	(10,16)
Regular UOV	1.981s	2.421s	934.305s	1205.271s
Circulant UOV	1.992s	2.395s	941.271s	1197.887s

So the complexity of UOV against UOV attack can be estimated by $q^{v-o-1}*o^4$, which is exponential in *v*-*o*. If *v* and *o* are close, UOV attack will be a powerful attack. If we choose $v \approx 2*o$ like regular UOV, the complexity of this attack will be exponential in *o*.

4.5 MinRank Attack

Rank attack is a powerful attack against many MPKC schemes. It can be divided into MinRank attack and HighRank attack. In MinRank attack [37] [38], attackers will conduct an exhaustive search to find a linear combination of the associated symmetric matrices P_i' of public key with minimal rank *r*. *r* is the minimal rank of the associated symmetric matrices F_i' of MPKC central map. As every submatrix A_i in F_i' is a randomly chosen *v*-by-*v* matrix in Circulant UOV, *r* will be larger than *v* with overwhelming probability. To attack the system, the problem becomes to search for a rank *v* matrix among linear combinations of *o* matrices of size *n*-by-*n*. As *o* is smaller than *v* and *n*, the best way to solve the problem is to use an exhaustive search [37], which takes $O(q^o n^3)$ operations in the base field.

4.6 HighRank Attack

The HighRank attack was designed to attack Rainbow or Rainbow-like MPKC schemes [34]. It can be considered to be the counterpart of the MinRank attack. The goal of this attack is to find a small kernel shared by a large number of linear combinations of P_i' and identify a fixed subspace $T^{-1}(O_u)$ of Rainbow lies in it. Then attackers can separate each Rainbow layer and generate signatures the same way as a legitimate user. The security of regular UOV is not threatened by the original HighRank attack because there are no such shared kernel in its central map.

However, Circulant UOV is vulnerable to HighRank attack if the parameter is not appropriately chosen. Let $P_{\mathrm{h}}' = \sum_{0}^{o-1} \lambda_i P_i'$ where λ_i are random elements in the base field. We can write

$$T^{-T} P_h' T^{-1} = \begin{pmatrix} A & B \\ B^T & 0 \end{pmatrix},$$

where A is a *v*-by-*v* matrix and B is a *v*-by-*o* matrix. Suppose matrix A is invertible, then we have $Rank(P_h')=Rank(A)+Rank(B)$. The difference between regular UOV and Circulant UOV is that matrix B in regular UOV is a random matrix but matrix B in Circulant UOV can be expressed as $B = B_1 \sum_{0}^{o-1} \lambda_i' R^i$, where λ_i' are random elements in the base field and R is the *o*-by-*o* rotating matrix

$$R=\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & 0 & 1 & 0 \\ 0 & \ddots & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

To analyze the security of Circulant UOV against HighRank attack, we have to analyze the rank behavior of matrix B. As B_1 is a v-by-o random matrix ($v \approx 2o$), $Rank(B_1)=o$ with overwhelming probability. Rank of matrix B is mainly depends on rank of $\sum_{0}^{o-1} \lambda_i' R^i$. The characteristic polynomial of matrix R is x^{o-1} over the base field. Then we can get the rank of $\sum_{0}^{o-1} \lambda_i' R^i$ is that

$$Rank(\sum_{0}^{o} \lambda_i' R^i) = o - degree(gcd(x^o - 1, \sum_{0}^{o-1} \lambda_i' x^i)).$$

For general q and o, attackers can use HighRank attack to find a group of λ_i' to make matrix B not full rank, which will leaks a subspace of $T^{-1}(O)$ with dimension o-$Rank(B)$. The complexity of finding a subspace with dimension d by using HighRank attack is about $O(q^d \cdot \frac{n^3}{6})$ .After finding enough small subspaces of $T^{-1}(O)$, attackers can combine these subspaces into a larger subspace U of $T^{-1}(O)$ and use U to find a preimage X of $Y=P(X)$ for any Y.

Suppose x^{o-1} can factor as $x^o - 1 = \sum_{k=0}^{t} f_k$ over the base field. Let $d_k=Degree(f_k)$, we can suppose that $d_0 \le d_1 \le \ldots \le d_t$. For each d_k, attackers can attack Circulant UOV as follows:

Step 1 Let $D_k=\{d_1,\ldots,d_k\}$.

Step 2 Get the first element d_j of D_k and form an arbitrary linear combination $P_h' = \sum_{i=0}^{o-1} \lambda_i P_i'$, Find $U_j=ker(P_h')$.

Step 3 If $dim(U_j) \ge 1$, set $(\sum_{i=0}^{o-1} \lambda_i P_i')U_j = 0$. If the solution set has dimension o-d_j, remove d_j from D_k. If $D_k \neq \emptyset$, goto Step 2.

Step 4 Let $o_k = \sum_{i=1}^{k} d_i$. Extend the basis of $U_0,\ldots,U_k$ to a basis for K^n and use it to transform the public polynomials into a Oil-Vinegar form with o_k Oil variables and n-o_k Vinegar variables.

Step 5 For a vector Y, randomly choose $v_1,..,v_{n-o_k} \in K$ and substitute $x_1,\ldots,x_{n-o_k}$ with $v_1,..,v_{n-o_k}$. Attackers will get o linear equations in o_k variables. With probability q^{o-o_k}, attackers can find a preimage X satisfies that $P(X)=Y$.

The complexity of this attack can be estimated by

$$HighRank(q,o)=\min_{d_k}\left(\frac{n^3}{6}\cdot(q^{d_1}+\cdots+q^{d_k}+q^{o-o_k})\right).$$

For different o, the complexity of HighRank attack varies widely. Take the base field GF(31) as an example, the complexity of HighRank attack is about 31^{28} for o=29 because $x^{29-1}=(x+30)(x^{28}+x^{27}+...+x+1)$ over GF(31). However, if o=28, we have $x^{28}-1=(x+1)(x+30)(x^2+1)(x^6+x^5+x^4+x^3+x^2+x+1)(x^6+10x^5+3x^4+10x^3+3x^2+10x+1)(x^6+21x^5+3x^4+21x^3+3x^2+21x+1)(x^6+30x^5+x^4+30x^3+x^2+30x+1)$ over GF(31). The complexity of HighRank attack is about 31^{16}. In order to prevent HighRank attack, we have to choose o carefully. HighRank attack makes the parameters of Circulant UOV less flexible.

4.7 Linearization Equation Attack

Linearization Equation attack is first discussed in [45] to break C* [51]. The core essence of Linearization Equation attack is to construct a potential linear relationship between the input and the output of MPKC public system. There is no evidence shows that UOV and Circulant UOV is vunlerable to Linearization Equation attack. To prove Circulant UOV is immune to this attack, we generate engouh input-ouput pairs of the public system of Circulant UOV, and then substitute them into the equations of Linearization Equation attack. The results show that there exist no linear relationship between input-output pairs of the public system of Circulant UOV. So the Linearization Equation attack cannot work on Circulant UOV.

4.8 Other Attacks

From the above analysis, we can conclude that Circulant UOV stands against all known attacks for UOV if we choose the parameter properly. One may think there may exist some special attacks which can exploit the rotating relations in the private key of Circulant UOV. In fact, rotating relations are hard to use in cryptanalysis of MPKC [15]. As we can see, there are only small parts of the private key has this rotating relations. It becomes even harder to exploit after applying the left affine transformation S to perturb the private key. Although Circulant UOV secure against all known attacks, we cannot give a provable security for Circulant UOV. Similarly, security of UOV and NTRU [48] depends on some hard problem, but does not reduce to them. There are MPKC schemes and lattice-based system which reduce to hard lattice problems, but these are much less efficient. We believe careful study of cryptanalytic techniques can determine the security of Circulant UOV.

5. Experiments and Comparisons

To demonstrate the efficiency of Circulant UOV, we implement it using AVX2 instructions on an Intel Core i7-4790 @3.60Ghz CPU. AVX2 is a SIMD instruction set for micro-processors from Intel. It expands most integer commands to 256-bit. We can pack 16 16-bit integer operands in its 256-bit ymm registers and do 16 integer operations per cycle. It can speed up matrix vector multiplication over 16-bit integers by a factor of 16 in theory. As there are many matrix vector multiplications in Circulant UOV encryption and decryption procedure, AVX2 instructions are extremely useful for Circulant UOV.

Nowadays, nearly all the computer support SIMD instructions. Traditional asymmetric cryptosystems such as RSA and ECC implemented in OpenSSL [39] have already taken the advantages of SIMD instructions. In CHES 2009, Chen et al. give a Streaming SIMD Extensions (SSE) implementation of MPKC on x86 CPUs [40]. It shows that the MPKC signature schemes are faster than the traditional asymmetric cryptosystems in both signing

time and verification time. However, they haven't implement UOV. In this paper, we implement Circulant UOV using SIMD integer operations in AVX2 instructions. We will give an overall comparison with other signature schemes.

We choose GF(31) as our base field. We use techniques mentioned in [40] and extend them from SSE version to AVX2 version. The technical details can be found in paper [40]. All the matrix vector multiplications and polynomial evaluations are vectorized.

5.1 The Minus Method

In [33], the authors presented UOV parameters to achieve 80, 100, 128 bits of security over the base field GF(31). From the discussion in Section 4, we know that Circulant UOV can use these parameters for the same security requirements to defend Direct attack. Unfortunately, parameters of UOV in [33] are not suitable for Circulant UOV because we have to make o slightly larger to prevent the HighRank attack and make L easily invertible. This will make the ration between v and o slightly smaller, as the complexity of the UOV attack can be estimated by $q^{v-o-1}*o^4$. Our choices can still meet the security requirements.

However, the constraint of o will make the public key size and verification time of Circulant UOV increase by 4% compared with that of regular UOV. In order to solve this problem. We introduce Minus method into our Circulant UOV. The Minus method was first suggested in [41] and discovered independently by Patarin and Matsumoto. The Minus method consists of deleting r polynomial components from a given multivariate public key. It was used to enhance the security of many MPKC schemes such as MI and HFE [42]. In Circulant UOV, we delete one or two public polynomials to make it have the same number of polynomials in public key as regular UOV. This method can make Circulant UOV have the same public key size and verification time with regular UOV. As the complexity of Direct attack depends on the number of public polynomials [33], this will not harm the security against Direct attack. After applying the Minus method, attackers have less information about the private key, so it will not make other attacks easier.

5.2 Compared with Regular UOV

After picking the appropriate parameters, we implement Circulant UOV with Minus method and UOV using AVX2 instructions. Then we measure the CPU cycles 1000 times for each parameters and calculate their average performance. We compare them in key generating time, signature generating time and private key size. The results are listed in **Table 4**. In the column of "Parameters ($\boldsymbol{o}$,$\boldsymbol{v}$,$\boldsymbol{r}$)", r denotes the number of polynomials we delete from the public key. As we have the same number of variables n and number of polynomials m in UOV and Circulant UOV in **Table 4**, the verifying time, public key size and signature length are the same. So we will not list them in the table.

From **Table 4**, we can observe that the private key size of Circulant UOV is reduced by about 45% compared with regular UOV. For better understanding, we present the signing time speed-up of Circulant UOV over UOV in **Fig. 3**.

Table 4. Comparison between Circulant UOV and regular UOV at different security levels over GF (31).

	Security (bit)	Parameters (o,v,r,m,n)	Private key Size (kB)	Signing time (10^3 cycles)	Key generating time (10^3 cycles)
UOV	80	(33,66,0,33,99)	96.5	1,893	44,964
	100	(41,82,0,41,123)	181.7	3,394	107,532
	128	(52,104,0,52,156)	364.9	4,093	299,160
Ours	80	(34,65,1,33,99)	53.9	126	149,472
	100	(43,80,2,41,123)	99.6	183	373,932
	128	(53,103,1,52,156)	196.5	277	732,888

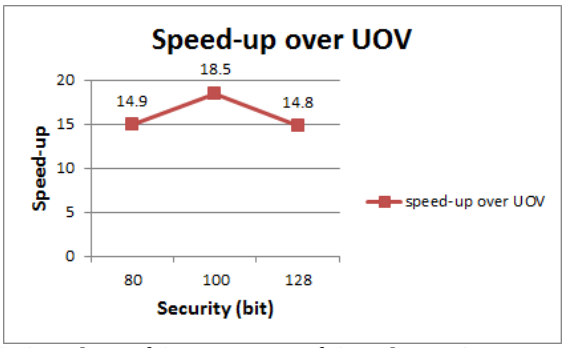

Fig. 3. Speed-up of Signing time of Circulant UOV over UOV.

Compared with UOV, the speed-up is top at 18.5. In our experiments, the Extended Euclidean algorithm in Circulant UOV is more than 25 times faster than Gauss Elimination in UOV. When o and v is small, the computational complexity of UOV signing algorithm is dominated by Gauss Elimination. This leads to a larger speed-up (80 bit and 100 bit security for examples). As o and v become larger, the proportion of the complexity of other parts become larger. The speed-up will be slightly reduced (128 bit security for example).

To show that Circulant UOV is energy efficient compared with regular UOV, we also implement signing process of Circulant UOV and UOV for 80 bit security in TelosB mote, which is a low power wireless sensor module. We ignore the verification process because Circulant UOV and UOV have the same verification process. We generate 100 instances for each scheme and record their average performance. The energy consumption can be calculated by the voltage, current and working time of TelosB. **Table 5** gives a comparison between them.

Table 5. Comparison between Circulant UOV and regular UOV on TelosB.

Scheme	Security (bit)	Code size (byte)	Signing time (ms)	Signing energy (mJ)
UOV	80	2218	809	4.37
Circulant UOV	80	2231	211	1.13

From **Table 5**, we can observe that Circulant UOV is energy efficient compared with regular UOV on TelosB mote. However, the speed-up of Circulant UOV over UOV in TelosB is smaller than that in a modern x64 CPU. This is because that the proportion of the complexity of matrix vector multiplications and polynomial evaluations becomes larger without the help of SIMD instructions.

5.3 Compared with other UOV variants

In this section, we compare our Circulant UOV with other UOV variants. In [33], authors implement Cyclic UOV in C on a laptop computer with a 2.53 GHz CPU. It's unfair to compare with our results directly, because we use the CPU hardware to accelerate the speed. As we both implement regular UOV, we can use it as a base to get Cyclic UOV results in our computing model.

Although Matrix-based method and NT method were originally designed for Rainbow, it can also be applied to UOV as UOV can be considered as a single layer Rainbow. We implement them using AVX2 instruction in our computer for better comparison. We compare Circulant UOV with the other UOV variants over GF(31) for 80 bit security (except Matrix-based UOV). The results are listed in **Table 6**.

The cells with better performance are marked as yellow. Circulant UOV is better at signing time and private key size; Cyclic UOV is better with respect to verification time and public key size; regular UOV is only better at key generating time.

Table 6. Comparing with other UOV variants over GF(31).

Scheme	Parameters (o,v)	Public key size (kB)	Private key size (kB)	Signature size (bit)	Key generating time(10^3cycles)	Signing time (10^3 cycles)	Verification time(10^3cycles)
UOV [26]	(33,66)	101.7	96.5	495	44,964	1,893	43
Cyclic UOV [33]	(33,66)	17.1	96.5	495	22,291,200	1,893	10
Matrix-based UOV [24]	(33,66,d=3)	101.7	54.1	495	151,284	1,242	43
NT UOV [24]	(33,66)	101.7	67.9	495	136,814	1,287	43
Ours	(34,65,r=1)	101.7	53.9	495	149,472	126	43

From **Table 6** we can conclude that Circulant method is better than Matrix-based method and NT method in terms of reducing the private key size and improving signing speed of UOV liked signature schemes.

As Rainbow can be considered as a multi-layer UOV, it's natural to think of extending our Circulant UOV to Circulant Rainbow. However, as parameters of each layer of Circulant Rainbow should satisfy the parameter constraints of Circulant UOV to prevent HighRank attack and make L easily invertible, this makes the public key size of Circulant Rainbow 1.5 times larger than regular Rainbow for 80 bit security over GF(31). In order to make the parameters of Circulant Rainbow more flexible we have to introduce some new techniques to it. We will focus on this in our future work.

5.4 Compared with Other Signature Schemes

Here we compare our Circulant UOV implementation with Rainbow, Gui, GLP, RSA and ECDSA implementations.

The fastest known MPKC signature schemes are Rainbow and Gui. For Gui, we get the implementation result from [13]. As there is no AVX implementation of Rainbow, we implement it using the same techniques as Circulant UOV. For lattice-based signatures, we choose GLP which enjoys software speed records for lattice-based signatures [49]. For RSA and ECDSA, we choose the parameters according to the latest NIST key management recommendation [43]. We run OpenSSL1.0.1t speed test program in our computer instead of implementing our own version. OpenSSL is the most popular crypto library in the world. It already takes advantage of SIMD instructions. Software is compiled for the x64 architecture, so the comparison will be reasonable. The overall results are listed in **Table 7**.

Table 7. Overall comparison with other signature schemes.

Scheme	Security (bit)	Public key size (Bytes)	Private key size (Bytes)	Signature size (bits)	Signing time (10^3 cycles)	Verification time (10^3 cycles)
Ours(GF(31),34,65,r=1)	80	101.7 K	53.9 K	495	126	43
Ours(GF(31),53,103,r=1)	128	393.6 K	196.5 K	568	306	277
Gui-96 (96,5,6,6) [13]	80	61.5 K	3.1 K	126	238	62
Gui-127 (127,9,4,6) [13]	120	139.2 K	5.2 K	163	1080	122
Rainbow(GF(31),24,20,20) [14]	80	57 K	37 K	320	197	35
RSA-1024 [39]	80	128	128	1024	475	54
RSA-4096 [39]	128	516	516	4096	27190	406
ECDSA B163 [39]	80	41	21	326	720	1440
ECDSA B283 [39]	128	70	35	556	1106	2160
GLP [49]	80	1536	256	9472	634	45

As we can see in **Table 7**, Circulant UOV outperforms all other signature schemes in signing time. Its verification speed outperforms all other signature schemes except Rainbow. However, the key size of Circulant UOV are larger than other signature schemes.

Anyway, the results we get are sound enough to show that Circulant UOV is comparable to other signature schemes. It is a promising candidate for Post-Quantum Cryptography.

6. Conclusion

In this paper, we propose Circulant UOV with shorter private key size and higher signature generating efficiency. We give some concrete parameters for different levels of security and make an overall comparison with other schemes to confirm the efficiency. Our experiments show that Circulant UOV is much faster than regular UOV in signing time, and it outperforms many other signature schemes in speed. It is a promising candidate for Post-Quantum Cryptography.

Here we list some directions for future work.

1) Use Circulant UOV to build UOV-based crypto schemes, such as Circulant Rainbow.
2) Use other primitives such as Toeplitz matrix instead of circulant matrix.
3) Examine the secure parameters in different base fields.

References

[1] P. Shor, "Algorithms for quantum computation: discrete logarithms and factoring," in *Proc. of Foundations of Computer Science, 1994 Proceedings, 35th Annual Symposium on. IEEE*, pp. 124–134, 1994. Article (CrossRef Link)

[2] P. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM Journal on Computing*, *vol. 26*, pp. 1484–1509, 1996. Article (CrossRef Link)

[3] D. J. Bernstein, J. Buchmann, and E. Dahmen, "Post-quantum cryptography," *Springer Science & Business Media*, 2009. Article (CrossRef Link)

[4] L. Chen, S. Jordan, Y.-K. Liu, D. Moody, R. Peralta, R. Perlner, and D. Smith-Tone, "Report on post-quantum cryptography," *National Institute of Standards and Technology Internal Report, vol. 8105*, 2016. Article (CrossRef Link)

[5] H. Imai and T. Matsumoto, "Algebraic methods for constructing asymmetric cryptosystems," in *Proc. of International Conference on Applied Algebra, Algebraic Algorithms, and Error-Correcting Codes. Springer*, pp. 108–119, 1985. Article (CrossRef Link)

[6] B. Yang and J. Chen, "Building secure tame-like multivariate public-key cryptosystems: The new TTS," *Information Security and Privacy. Springer*, pp. 518–531, 2005. Article (CrossRef Link)

[7] D. Gligoroski, S. Markovski, and S. J. Knapskog, "Multivariate quadratic trapdoor functions based on multivariate quadratic quasigroups," in *Proc. of Proceedings of the American Conference on Applied Mathematics, Stevens Point, Wisconsin, USA, World Scientific and Engineering Academy and Society (WSEAS)*, pp. 44–49, 2008. Article (CrossRef Link)

[8] C. Tao, A. Diene, S. Tang, and J. Ding, "Simple Matrix Scheme for Encryption," *PQCrypto*, vol. 13, pp. 231–242, 2013. Article (CrossRef Link)

[9] J. Patarin, "Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms," in *Proc. of Advances in Cryptology-EUROCRYPT' 96. Springer*, pp. 33–48, 1996. Article (CrossRef Link)

[10] J. Porras, J. Baena, and J. Ding, "ZHFE, a new multivariate public key encryption scheme," *Post-Quantum Cryptography. Springer*, pp. 229–245, 2014. Article (CrossRef Link)

[11] J. Ding, B. Yang, C. Chen, M. Chen, and C. Cheng, "New differential algebraic attacks and reparametrization of Rainbow," in *Proc. of Proceedings of the 6th international conference on Applied cryptography and network security*. *Springer Verlag*, pp. 242–257, 2008. Article (CrossRef Link)
[12] E. Thomae, "A generalization of the Rainbow Band Separation attack and its applications to multivariate schemes," *IACR Cryptology ePrint Archive*, vol. 2012, p. 223, 2012. Article (CrossRef Link)
[13] A. Petzoldt, M.-S. Chen, B.-Y. Yang, C. Tao, and J. Ding, "Design principles for HFEv-based multivariate signature schemes," in *Proc. of International Conference on the Theory and Application of Cryptology and Information Security*. *Springer*, pp. 311–334, 2015. Article (CrossRef Link)
[14] J. Ding and D. Schmidt, "Rainbow, a new multivariable polynomial signature scheme," *Applied Cryptography and Network Security*. *Springer*, pp. 317–366, 2005. Article (CrossRef Link)
[15] A. Petzoldt, S. Bulygin, and J. Buchmann, "CyclicRainbow–A multivariate signature scheme with a partially cyclic public key," *Progress in Cryptology-INDOCRYPT 2010*. *Springer*, pp. 33–48, 2010. Article (CrossRef Link)
[16] 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. Article (CrossRef Link)
[17] A. Petzoldt and S. Bulygin, "Linear recurring sequences for the UOV key generation revisited," *Information Security and Cryptology–ICISC 2012*. *Springer*, pp. 441–455, 2013. Article (CrossRef Link)
[18] A. Petzoldt, S. Bulygin, and J. Buchmann, "Fast verification for improved versions of the UOV and Rainbow signature schemes," *Post-Quantum Cryptography*. *Springer*, pp. 188–202, 2013. Article (CrossRef Link)
[19] H. Seo, J. Kim, J. Choi, T. Park, Z. Liu, and H. Kim, "Small private key MQPKS on an embedded microprocessor," *Sensors*, vol. 14, no. 3, pp. 5441–5458, 2014. Article (CrossRef Link)
[20] F. Borges, A. Petzoldt, and R. Portugal, "Small private keys for systems of multivariate quadratic equations using symmetric cryptography," *Available online: http://www.informatik.tu-darmstadt.de/fileadmin/userupload/GroupTK/UOVcnmac2012-final.pdf* (accessed on 10 January 2014), 2014. Article (DirectLink)
[21] T. Yasuda, T. Takagi, and K. Sakurai, "Efficient variant of Rainbow using sparse secret keys," *Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA)*, vol. 5, no. 3, pp. 3–13, 2014. Article (CrossRef Link)
[22] T. Yasuda, J. Ding, T. Takagi, and K. Sakurai, "A variant of Rainbow with shorter secret key and faster signature generation," in *Proc. of Proceedings of the first ACM workshop on Asia public key cryptography*. *ACM*, pp. 57–62, 2013. Article (CrossRef Link)
[23] T. Yasuda, T. Takagi, and K. Sakurai, "Efficient variant of Rainbow without triangular matrix representation," in *Proc. of Information and Communication Technology-EurAsia Conference. Springer*, pp. 532–541, 2014. Article (CrossRef Link)
[24] Y. Tan and S. Tang, "Two approaches to build UOV variants with shorter private key and faster signature generation," in *Proc. of International Conference on Information Security and Cryptology*. *Springer, 2015, pp.* 57–74. Article (CrossRef Link)
[25] A. Kipnis and A. Shamir, "Cryptanalysis of the Oil and Vinegar signature scheme," in *Proc. of Annual International Cryptology Conference. Springer*, pp. 257–266, 1998. Article (CrossRef Link)
[26] A. Kipnis, J. Patarin, and L. Goubin, "Unbalanced Oil and Vinegar signature schemes," *Advances in Cryptology-EUROCRYPT'99*. *Springer*, pp. 206–222, 1999. Article (CrossRef Link)
[27] D. Bini, G. M. Del Corso, G. Manzini, and L. Margara, "Inversion of circulant matrices over Zm," *Automata, Languages and Programming. Springer*, pp. 719–730, 1998. Article (CrossRef Link)
[28] W. Bosma, J. Cannon, and C. Playoust, "The Magma algebra system I: The user language," *Journal of Symbolic Computation*, vol. 24, no. 3, pp. 235–265, 1997. Article (CrossRef Link)
[29] A. Joux and V. Vitse, "A variant of the F4 algorithm," *Topics in Cryptology–CT-RSA 2011*. *Springer*, pp. 356–375, 2011. Article (CrossRef Link)

[30] J.-C. Faugere, "A new efficient algorithm for computing Gröbner bases (F4)," *Journal of pure and applied algebra*, vol. 139, no. 1, pp. 61–88, 1999. Article (CrossRef Link)
[31] B. Sturmfels, "What is a Gröbner basis," *Notices Amer. Math. Soc*, vol. 52, no. 10, pp. 1199–1200, 2005. Article (DirectLink)
[32] L. Bettale, J.-C. Faugere, and L. Perret, "Hybrid approach for solving multivariate systems over finite fields," *Journal of Mathematical Cryptology*, vol. 3, no. 3, pp. 177–197, 2009. Article (CrossRef Link)
[33] A. Petzoldt, "Selecting and reducing key sizes for multivariate cryptography," *Doctoral dissertation, tuprints*, 2013. Article (Direct Link)
[34] J. Ding, B.-Y. Yang, C.-H. O. Chen, M.-S. Chen, and C.-M. Cheng, "New differential-algebraic attacks and reparametrization of Rainbow," in *Proc. of International Conference on Applied Cryptography and Network Security. Springer*, pp. 242–257, 2008. Article (CrossRef Link)
[35] M. Bardet, J.-C. Faugere, and B. Salvy, "On the complexity of Gröbner basis computation of semi-regular overdetermined algebraic equations," in *Proc. of Proceedings of the International Conference on Polynomial System Solving*, pp. 71–74, 2004. Article (DirectLink)
[36] M. Bardet, J.-C. Faugere, B. Salvy, and B.-Y. Yang, "Asymptotic expansion of the degree of regularity for semi-regular systems of equations," *Mega*, pp. 1–14, 2005. Article (CrossRef Link)
[37] J.-C. Faugere, F. Levy-Dit-Vehel, and L. Perret, "Cryptanalysis of MinRank," i*Advances in Cryptology–CRYPTO 2008. Springer*, pp. 280–296, 2008. Article (CrossRef Link)
[38] O. Billet and H. Gilbert, "Cryptanalysis of Rainbow," in *Proc. of International Conference on Security and Cryptography for Networks. Springer*, pp. 336–347, 2006. Article (CrossRef Link)
[39] "OpenSSL,". Article (DirectLink)
[40] A. I.-T. Chen, M.-S. Chen, T.-R. Chen, C.-M. Cheng, J. Ding, E. L.-H. Kuo, F. Y.-S. Lee, and B.-Y. Yang, "SSE implementation of multivariate PKCs on modern x86 CPUs," *Cryptographic Hardware and Embedded Systems-CHES 2009. Springer*, pp. 33–48, 2009. Article (CrossRef Link)
[41] A. Shamir, "Efficient signature schemes based on birational permutations," in *Proc. of Annual International Cryptology Conference. Springer*, pp. 1–12, 1993. Article (CrossRef Link)
[42] J. Patarin, L. Goubin, and N. Courtois, "C-+* and HM: Variations around two schemes of T. Matsumoto and H. Imai," in *Proc. of International Conference on the Theory and Application of Cryptology and Information Security. Springer*, pp. 35–50, 1998. Article (CrossRef Link)
[43] E. Barker, W. Barker, W. Burr, W. Polk, M. Smid, P. D. Gallagher et al., "NIST special publication 800-57 recommendation for key management–part 1: General," 2012. Article (CrossRef Link)
[44] Courtois and Nicolas T, "The security of hidden field equations HFE," in *Proc. of Track at the RSA Conference. Springer*, pp. 266–281, 2001. Article (CrossRef Link)
[45] Patarin and Jacques, "Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88," in *Proc. of Annual International Cryptology Conference. Springer*, pp. 248–261, 1995. Article (CrossRef Link)
[46] V. Dubois, P. Fouque, A. Shamir and J. Stern, "Practical Cryptanalysis of SFLASH," in *Proc. of Annual International Cryptology Conference. Springer*, pp. 1–12, 2007. Article (CrossRef Link)
[47] R. Perlner and D. Smith-Tone, "A classification of differential invariants for multivariate post-quantum cryptosystems," in *Proc. of International Workshop on Post-Quantum Cryptography. Springer*, pp. 165–173, 2013. Article (CrossRef Link)
[48] J. Hoffstein, J. Pipher and J. Silverman, "NTRU: A ring-based public key cryptosystem," *Algorithmic number theory. Springer*, pp. 267–288, 1998. Article (CrossRef Link)
[49] T. Guneysu, T. Oder, T. Pöppelmann and P. Schwabe, "Software speed records for lattice-based signatures," in *Proc. of International Workshop on Post-Quantum Cryptography. Springer*, pp. 67–82, 2013. Article (CrossRef Link)
[50] A. Szepieniec, J. Ding and B. Preneel, "Extension Field Cancellation: a New Central Trapdoor for Multivariate Quadratic Systems," in *Proc. of International Workshop on Post-Quantum Cryptography. Springer*, pp. 182–196, 2016. Article (CrossRef Link)

[51] T. Matsumoto and H. Imai, "Public quadratic polynomial-tuples for efficient signature-verification and message-encryption," in *Proc. of Workshop on the Theory and Application of of Cryptographic Techniques. Springer*, pp. 419–453, 1988. Article (CrossRef Link)

Appendix

Table of Notations

Notation	Meaning
o	Number of Oil variables
v	Number of Vinegar variables
n	Number of variables
m	Number of polynomials
q	Order of the base field
r	Number of polynomials deleted from the public key
K	Base field
P	Public system of Circulant UOV
F	Central map of Circulant UOV
T	An affine transformation
S	An affine transformation
A_i	v-by-v matrix standing for Vinegar-Vinegar cross-terms
B_i	v-by-o matrix standing for Oil-Vinegar cross-terms
β_i	Linear coefficients of Oil variables
α_i	Linear coefficients of Vinegar variables
c_i	Constant term
$Z_p[x]$	Univariate polynomial ring over Z_p
$GF(q)$	Galois field of order q
d_{reg}	Degree of regularity
F_i'	Associated symmetric matrix of the central map
P_i'	Associated symmetric matrix of the public system
O	Oil space
$Rank(\cdot)$	Rank of a matrix
$Degree(\cdot)$	Degree of a polynomial
$Gcd(\cdot,\cdot)$	Greatest common divisor of two polynomials
Rainbow	A MPKC signature scheme
Rainbow-Band-Separation attack	An attack over MPKC
SIMD	Single Instruction Multiple Data
AVX	Advanced Vector Extensions
PRNG	Pseudorandom Number Generator

Zhiniang Peng received the BE Degree from South China University of Technology in 2013. Now he is a doctoral candidate at the School of Computer Science and Engineering, South China University of Technology. His current research interests include lattice-based cryptography, crypto chip design, and multivariate public key cryptography.

Shaohua Tang received the B.Sc. and M.Sc. Degrees in applied mathematics from South China University of Technology, China, in 1991 and 1994, respectively, and the Ph.D. Degree in communication and information system from South China University of Technology, in 1998. He was a visiting scholar with North Carolina State University, USA, and a visiting professor with University of Cincinnati, USA. He has been a full professor with the School of Computer Science and Engineering, South China University of Technology since 2004. His current research interests include information security, data security and privacy preserving in cloud computing and big data. He has authored or co-authored over 100 technical papers in journals and conference proceedings. He is a member of the IEEE and the IEEE Computer Society.