Computer Security Symposium 2021
26 - 29 October 2021

A New Fault Attack on UOV Multivariate Signature Scheme

Hiroki Furue[1] Yutaro Kiyomura[2] Tatsuya Nagasawa[1] Tsuyoshi Takagi[1]

Abstract: The unbalanced oil and vinegar signature scheme (UOV), which is one of the multivariate signature schemes, is expected to be secure against quantum attacks. In this paper, we propose a new fault attack on UOV using, for the first time, faults caused on a central map. In the proposed attack, the linear map T of the secret key is partially recovered using signatures generated from a faulty secret key. Furthermore, we propose a new algebraic method for executing a known attack with a smaller complexity by using the partially recovered information of T. For a parameter set UOV(16,60,39) satisfying 100-bit security, our simulation shows that the proposed attack recovers the secret key with a smaller complexity than the claimed security level with approximately 90% probability.

Keywords: post-quantum cryptography, multivariate cryptography, UOV, fault attack

1. Introduction

Currently used public key cryptosystems such as RSA and ECC can be broken in polynomial time using a quantum computer executing Shor's algorithm [12]. Thus, there has been growing interest in post-quantum cryptography (PQC), which is secure against quantum computing attacks. The amount of research conducted on PQC has thus been accelerating, and the U.S. National Institute for Standards and Technology (NIST) has initiated a PQC standardization project [10].

Multivariate public key cryptography (MPKC), based on the difficulty of solving a system of multivariate quadratic polynomial equations over a finite field (the multivariate quadratic ($\mathcal{MQ}$) problem), is regarded as a strong candidate for PQC. The $\mathcal{MQ}$ problem is NP-complete [5] and is thus likely to be secure in the post-quantum era.

The unbalanced oil and vinegar signature scheme (UOV) [7], a multivariate signature scheme proposed by Kipnis et al. at EUROCRYPT 1999, has withstood various types of attacks during a period of approximately 20 years. UOV is a well-established signature scheme owing to its short signature and short execution time. Rainbow [3], a multilayer UOV variant, was selected as a third-round finalist in the NIST PQC project [11].

However, there have been few studies on physical attacks against UOV or Rainbow. In [6], Hashimoto et al. proposed two fault attacks on MPKCs. The first attack uses faults that change the coefficients of unknown terms in the central quadratic map. This attack works on MPKCs whose public key $P = S \circ F \circ T$ is composed of two linear maps S, T and a central map F, such as Rainbow. However, because the secret key of UOV does not include a linear map S, this attack does not work on this scheme. The other fault attack in [6] is one in which the attacker fixes parameters chosen at random during the signature generation step.

Furthermore, in [9], Mus et al. proposed a fault attack on LUOV [2], a variant of UOV, using a subfield. This attack utilizes the secret key in the binary field and thus does not work on the plain UOV.

Our Contribution

In this study, we propose a new fault attack on UOV. To the best of our knowledge, this is the first fault attack on UOV that applies faults generated on the secret key. In the proposed attack, we assume that the attacker randomly changes the coefficient of the secret key F, T, and that the fault is permanent. It should be noted that the attacker cannot know the location of the fault.

The proposed attack first partially recovers the information of T using faults caused on the central map F. For each fault, by using some pairs of signatures and the difference in messages generated by true and faulty keys, we recover some row vectors of T. These manipulations are iterated unless a new fault is caused on T. Subsequently, by using partially recovered information, we reduce the given public key system to a smaller UOV public key system. As a result, the UOV attack can be executed on a reduced system with a smaller complexity.

We simulate the proposed attack on two parameter sets, UOV(16,60,39) and UOV(256,50,33) satisfying 100-bit security, and estimate the complexity of the UOV attack on the resulting system. As a result, in UOV(16,60,39) and UOV(256,50,33), the proposed attack is executed with a

[1] The University of Tokyo
[2] NTT Social Informatics Laboratories

smaller complexity than the claimed security level with approximately 90% and 50% probability, respectively.

Organizations

In Section 2, we describe the construction of the multivariate signature schemes, UOV, and the UOV attack. In Section 3, we detail the proposed attack and its complexity. In Section 4, we show how the proposed attack reduces the complexity of recovering the secret key on concrete parameter sets. Finally, we provide some concluding remarks in Section 5.

2. Preliminaries

In this section, we first describe the $\mathcal{MQ}$ problem and general signature schemes based on this problem. Subsequently, we review the construction of UOV [7]. There are four currently known attacks on UOV: the direct attack, UOV attack [8], reconciliation attack [4], and intersection attack [1]. The direct attack obtains a signature for a given message directly, whereas the other three attacks recover the secret key. In the attack proposed in Section 3, the UOV attack [8] is utilized to estimate the complexity of the proposed attack. Thus, we briefly describe the UOV attack in Subsection 2.3.

2.1 Multivariate Signature Schemes

Let $\mathbb{F}_q$ be a finite field with q elements, and let n and m be two positive integers. For a system of quadratic polynomials $P = (p_1(x_1, \dots, x_n), \dots, p_m(x_1, \dots, x_n))$ in n variables over $\mathbb{F}_q$, the problem of obtaining a solution $\mathbf{x} \in \mathbb{F}_q^n$ to $P(\mathbf{x}) = \mathbf{0}$ is called the $\mathcal{MQ}$ problem. Garey and Johnson [5] proved that this problem is NP-complete if $n \approx m$, and thus it is considered to have the potential to resist quantum computer attacks.

Next, we briefly describe the construction of the general multivariate signature schemes. First, an easily invertible quadratic map $F = (f_1, \dots, f_m) : \mathbb{F}_q^n \to \mathbb{F}_q^m$, called a *central map*, is generated. Next, two invertible linear maps $T : \mathbb{F}_q^n \to \mathbb{F}_q^n$ and $S : \mathbb{F}_q^m \to \mathbb{F}_q^m$ are randomly chosen to hide the structure of F. These two linear maps S and T can be seen as two matrices in $\mathbb{F}_q^{m\times m}$ and $\mathbb{F}_q^{n\times n}$. The public key P is then provided as a polynomial map:

$$P = S \circ F \circ T : \mathbb{F}_q^n \to \mathbb{F}_q^m. \tag{1}$$

The secret key comprises S, F, and T. The signature is generated as follows: Given a message $\mathbf{m} \in \mathbb{F}_q^m$ to be signed, compute $\mathbf{m}_1 = S^{-1}(\mathbf{m})$, and obtain a solution $\mathbf{m}_2$ to the equation $F(\mathbf{x}) = \mathbf{m}_1$. This gives the signature $\mathbf{s} = T^{-1}(\mathbf{m}_2) \in \mathbb{F}_q^n$ for the message. Verification is applied by confirming whether $P(\mathbf{s}) = \mathbf{m}$.

2.2 Unbalanced Oil and Vinegar Signature Scheme

Let v be a positive integer and $n = v + m$. For variables $\mathbf{x} = (x_1, \dots, x_n)$ over $\mathbb{F}_q$, we call $x_1, \dots, x_v$ *vinegar variables* and $x_{v+1}, \dots, x_n$ *oil variables.* In the UOV scheme, a central map $F = (f_1, \dots, f_m) : \mathbb{F}_q^n \to \mathbb{F}_q^m$ is designed such that each f_k $(k = 1, \dots, m)$ is a quadratic polynomial of the form

$$f_k(x_1, \dots, x_n) = \sum_{i=1}^{v} \sum_{j=i}^{n} \alpha_{i,j}^{(k)} x_i x_j \tag{2}$$

where $\alpha_{i,j}^{(k)} \in \mathbb{F}_q$. A linear map $T : \mathbb{F}_q^n \to \mathbb{F}_q^n$ is randomly chosen. Next, the public key map $P : \mathbb{F}_q^n \to \mathbb{F}_q^m$ is computed using $P = F \circ T$. The linear map S in equation (1) is not required because it does not help hide the structure of F in UOV. Thus, the secret key is composed of F and T.

Next, we describe the inversion of the central map F. Given $\mathbf{y} \in \mathbb{F}_q^m$ as a message, random values $a_1, \dots, a_v$ in $\mathbb{F}_q$ are chosen as the values of the vinegar variables. We can then efficiently obtain a solution $(a_{v+1}, \dots, a_n)$ for the equation $F(a_1, \dots, a_v, x_{v+1}, \dots, x_n) = \mathbf{y}$ because this is a linear system of m equations in m oil variables. If there is no solution to this equation, we choose new random values $a'_1, \dots, a'_v$, and repeat the procedure. Eventually, we obtain the solution $\mathbf{x} = (a_1, \dots, a_v, a_{v+1}, \dots, a_n)$ to $F(\mathbf{x}) = \mathbf{y}$. In this manner, we execute the signing process described in Subsection 2.1.

2.3 UOV Attack

The UOV attack [8] obtains a linear map $T' : \mathbb{F}_q^n \to \mathbb{F}_q^n$ such that every component of $F' = P \circ T'^{-1}$ has the form of equation (2). Such a T' is called an *equivalent key.* The UOV attack obtains the subspace $T^{-1}(\mathcal{O})$ of $\mathbb{F}_q^n$, where $\mathcal{O}$ is the oil subspace defined as

$$\mathcal{O} := \left\{ (0, \dots, 0, \alpha_1, \dots, \alpha_m)^\top \;\middle|\; \alpha_i \in \mathbb{F}_q \right\}.$$

This subspace $T^{-1}(\mathcal{O})$ can induce an equivalent key. To obtain $T^{-1}(\mathcal{O})$, the UOV attack chooses two invertible matrices W_i, W_j from the set of linear combinations of the representation matrices of $p_1, \dots, p_m$. It then probabilistically recovers a part of the subspace $T^{-1}(\mathcal{O})$ by computing the invariant subspace of $W_i^{-1} W_j$. The complexity of the UOV attack is estimated to be

$$O\left(q^{v-m-1} \cdot m^4\right).$$

3. New Fault Attack on UOV

In this section, we propose a new fault attack on UOV that utilizes faults caused on the central map. The proposed attack mainly consists of three steps: First, some rows of the secret key T are recovered using faults. Second, T is transformed. Third, the UOV attack is executed on the transformed system with a smaller complexity. For the proposed attack, we assume that the attacker causes a permanent fault, thereby changing the coefficients of the secret key F and T.

3.1 Recovery of Information of Secret Key $\boldsymbol{T}$

In this subsection, we describe a method for partially recovering the secret key T by utilizing a fault caused on the central map F. During the proposed attack, we assume that the attacker randomly changes the coefficient of secret key F, T, and the fault is permanent. Note that the attacker cannot know the location of the fault.

Now, we consider the case in which the first fault is caused on F and changes the coefficient $\alpha_{i,j}^{(k)}$ $(1 \leq i \leq j \leq n, 1 \leq k \leq m)$ in equation (2) into $\bar{\alpha}_{i,j}^{(k)}$ $(\alpha_{i,j}^{(k)} \neq \bar{\alpha}_{i,j}^{(k)})$. Then, the faulty central map is denoted by F'.

First, we generate $n(n+1)/2$ pairs of a signature and the difference between the two messages from the faulty and true keys. These pairs are generated through the following three steps: First, a message $\mathbf{m}_\ell \in \mathbb{F}_q^m$ is randomly chosen. Second, a faulty signature $\mathbf{s}_\ell \in \mathbb{F}_q^n$ for the message $\mathbf{m}_\ell$ is obtained using the signing oracle of the faulty secret key. Third, we find the difference between δ_ℓ of $P(\mathbf{s}_\ell)$ and $\mathbf{m}_\ell$. These steps are described as follows:

(1) $\mathbf{m}_\ell \in \mathbb{F}_q^m$
(2) $\mathbf{s}_\ell = T^{-1} \circ {F'}^{-1}(\mathbf{m}_\ell)$
(3) $\delta_\ell = P(\mathbf{s}_\ell) - \mathbf{m}_\ell$,

These manipulations are iterated $n(n+1)/2$ times.

We then hold the following:

$$\begin{aligned}\delta_\ell &= (F \circ T - F' \circ T)(\mathbf{s}_\ell) \\ &= (F - F') \circ T(\mathbf{s}_\ell) \\ &= \left(0, \ldots, 0, \left(\alpha_{i,j}^{(k)} - \bar{\alpha}_{i,j}^{(k)}\right) \cdot T^{(i)}(\mathbf{s}_\ell) \cdot T^{(j)}(\mathbf{s}_\ell), 0, \ldots, 0\right),\end{aligned}$$

where $T^{(i)}(\cdot)$ and $T^{(j)}(\cdot)$ denote the i-th and j-th elements of $T(\cdot)$, respectively. This shows that when $T^{(i)}(\mathbf{s}_\ell) \neq 0$ and $T^{(j)}(\mathbf{s}_\ell) \neq 0$, δ_ℓ has the only nonzero element as the k-th element. Let $\delta_\ell = (\delta_1^{(\ell)}, \ldots, \delta_m^{(\ell)})^\top$, $\mathbf{s}_\ell = (s_1^{(\ell)}, \ldots, s_n^{(\ell)})^\top$, and $T_{i,j}$ be the i, j-element of T. We then hold

$$\begin{aligned}\delta_k^{(\ell)} &= \left(\alpha_{i,j}^{(k)} - \bar{\alpha}_{i,j}^{(k)}\right) \cdot \left(\sum_p t_{i,p} s_p^{(\ell)}\right) \cdot \left(\sum_r t_{j,r} s_r^{(\ell)}\right) \\ &= \left(\alpha_{i,j}^{(k)} - \bar{\alpha}_{i,j}^{(k)}\right) \sum_{p \leq r} s_p^{(\ell)} s_r^{(\ell)} \begin{cases} (t_{i,p} t_{j,r} + t_{j,p} t_{i,r}) \\ \qquad\qquad (p \neq r) \\ t_{i,p} t_{j,p} \\ \qquad\qquad (p = r) \end{cases}.\end{aligned}$$

Now, we introduce new $n(n+1)/2$ variables $\{y_{p,r}\}_{1 \leq p \leq r \leq n}$, where every component $y_{p,r}$ corresponds to $(s_{ip} s_{jr} + s_{jp} s_{ir})$ in the case of $p \neq r$, and $s_{ip} s_{jp}$ in the case of $p = r$. We then generate a linear system of $n(n+1)/2$ equations in $n(n+1)/2$ variables $\{y_{p,r}\}_{1 \leq p \leq r \leq n}$ as follows:

$$\sum_{p \leq r} s_p^{(\ell)} s_r^{(\ell)} y_{p,r} = \delta_k^{(\ell)} \qquad \left(1 \leq \ell \leq \frac{n(n+1)}{2}\right). \quad (3)$$

By solving this linear system, the values of $y_{1,1}, \ldots, y_{n,n}$ are determined uniquely with a high probability. If the linear system has some solutions, we then add a new pair of faulty signatures and the difference between the two messages until reaching the solution is uniquely determined.

Subsequently, from the values obtained for $\{y_{p,r}\}_{1 \leq p \leq r \leq n}$, we generate two vectors $(a_1, \ldots, a_n)$ and $(b_1, \ldots, b_n)$ satisfying

$$\begin{cases} a_i b_j + a_j b_i = y_{i,j} \quad (i < j) \\ a_i b_i = y_{i,i} \end{cases}, \qquad (4)$$

as in the definition of $\{y_{p,r}\}_{1 \leq p \leq r \leq n}$. This can be easily executed by solving some small systems. These two vectors correspond to constant multiples of two row vectors of T, such as $(c_1 t_{i,1}, \ldots, c_1 t_{i,n})$ and $(c_2 t_{j,1}, \ldots, c_2 t_{j,n})$, where $c_1 \cdot c_2 = \alpha_{i,j}^{(k)} - \bar{\alpha}_{i,j}^{(k)}$.

After executing manipulations for the first fault described above, we cause another fault and recover two row vectors of T by using a similar method if the new fault is also caused on F. The main difference from the first fault is that $\delta^{(\ell)}$ may have several nonzero elements. However, by subtracting $\left(\sum_{p \leq r} s_p^{(\ell)} s_r^{(\ell)} y_{p,r}\right) \mathbf{e}_k$ from $\delta^{(\ell)}$ for every set of $\{y_{p,r}\}_{1 \leq p \leq r \leq n}$ ($\mathbf{e}_k$ denotes the k-th unit vector), it becomes a vector with one nonzero element, as in the above case. Note that if a recovered row vector of T is dependent on one of the row vectors already recovered, then this means that the same vector is recovered in a duplicate manner.

These manipulations are iterated until a new fault is caused by T, which can be easily confirmed because $\delta^{(\ell)}$ has many nonzero elements after subtracting $\left(\sum_{p \leq r} s_p^{(\ell)} s_r^{(\ell)} y_{p,r}\right) \mathbf{e}_k$. Algorithm 1 describes this step algorithmically.

Algorithm 1 Recovering row vectors of T

```
$T_{row} = \phi$
$a = 0$
while $|T_{row}| < n$ do
    cause a new fault
    $a \leftarrow a + 1$
    for $\ell = 1, \ldots, n(n+1)/2$ do
        $\mathbf{m}_\ell \leftarrow \mathbb{F}_q^m$
        $\mathbf{s}_\ell \leftarrow S^{-1} \circ F'^{-1}(\mathbf{m}_\ell)$
        $\delta_\ell \leftarrow P(\mathbf{s}_\ell) - \mathbf{m}_\ell - \sum_{a'} \left(\sum_{p \leq r} s_p^{(\ell)} s_r^{(\ell)} y_{p,r}^{(a')}\right) \mathbf{e}_{k_{a'}}$
    end for
    if only one element ($k$-th one) of $\delta_\ell$ is nonzero then
        $k_a \leftarrow k$
    else
        break while
    end if
    $y_{p,r}^{(a)} \leftarrow$ Solve equations (3) for $y_{p,r}$
    Find $(a_1, \ldots, a_n), (b_1, \ldots, b_n)$ satisfying equation (4)
    if $(a_1, \ldots, a_n)$ is independent from each element of $T_{row}$
    then
        $T_{row} \leftarrow T_{row} \cup (a_1, \ldots, a_n)$
    end if
    if $(b_1, \ldots, b_n)$ is independent from each elements of $T_{row}$
    then
        $T_{row} \leftarrow T_{row} \cup (b_1, \ldots, b_n)$
    end if
end while
```

3.2 Reduction to Smaller UOV

In this subsection, we describe how to reduce the public key system into a smaller UOV public key by using constant multiples of row vectors of T obtained in Subsection 3.1. The first part of this manipulation is mainly originated from [6]. Herein, we assume that the $i_1, \ldots, i_\alpha$-th row vectors of T

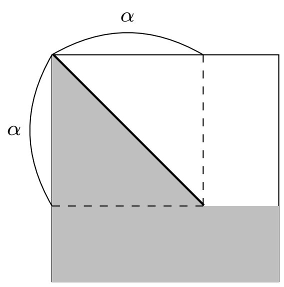

Fig. 1 An example of $T \cdot T_1 \cdots T_\alpha$ (the white part is zero.)

are recovered, as described in Subsection 3.1 (although the attacker does not know which vectors of T are obtained).

First, we choose one row vector $(a_1, \dots, a_n)$ obtained in Subsection 3.1 and take the matrix T_1 transforming the row vector of T corresponding $(a_1, \dots, a_n)$. We then choose one nonzero element a_{k_1} from $(a_1, \dots, a_n)$ and take T_1 such that the k_1-th row vector is

$$\left(-\frac{a_1}{a_{k_1}}, \dots, -\frac{a_{k_1-1}}{a_{k_1}}, 1, -\frac{a_{k_1+1}}{a_{k_1}}, \dots, -\frac{a_n}{a_{k_1}}\right),$$

and the other k'-th row vectors are $\mathbf{e}_{k'}^\top$. For example, in the case in which $k_1 = 1$, T_1 has the following form:

$$\begin{pmatrix} 1 & -\frac{a_2}{a_1} & \cdots & -\frac{a_n}{a_1} \\ 0 & & & \\ \vdots & & I_{n-1} & \\ 0 & & & \end{pmatrix}.$$

Then, by multiplying T_1 to T from the right side, the elements of the row vector corresponding to $(a_1, \dots, a_n)$ become zero except for the k_1-th element.

We iterate such processes for all α row vectors obtained in Subsection 3.1. For the i-th row vector $(b_1, \dots, b_n)$ of the α row vectors, we first compute

$$(b'_1, \dots, b'_n) = (b_1, \dots, b_n) \cdot T_1 \cdots T_{i-1}.$$

We then choose one nonzero element b'_{k_i} such that $k_i \notin \{k_1, \dots, k_{i-1}\}$ and take a matrix T_i such that the diagonal elements are 1 and $(T_i)_{k_i,j}$ $(j \notin \{k_1, \dots, k_i\})$ is $-b'_j/b'_{k_i}$.

By executing these steps, we obtain α matrices $T_1, \dots, T_\alpha$. For example, when $(i_1, \dots, i_\alpha) = (1, \dots, \alpha)$ and $(k_1, \dots, k_\alpha) = (1, \dots, \alpha)$, $T \cdot T_1 \cdots T_\alpha$ has the form shown in Figure 1.

Subsequently, we describe a method for reducing the public key to a smaller UOV system by using $T_1, \dots, T_\alpha$. Here, $[n]$ $(n \in \mathbb{N})$ denotes a set $\{1, \dots, n\}$, and for an $a \times b$ matrix A, $I \subseteq [a]$, and $J \subseteq [b]$, $A[I, J]$ denotes the submatrix of A given by the row indices in I and the column indices in J. We also let $T' = T \cdot T_1 \cdots T_\alpha$.

We substitute 0 for $x_{k_1}, \dots, x_{k_\alpha}$ of the transformed public key $P \circ (T_1 \cdots T_\alpha) = F \circ T'$. If we denote the resulting public key system in the remaining $n - \alpha$ variables by $\bar{P}$, then $\bar{P}$ is constructed by composing F and a linear map represented by $T'[[n], J]$, where $J = [n] \setminus \{k_1, \dots, k_\alpha\}$. As shown in Figure 1, the $i_1, \dots, i_\alpha$-th row vectors of $T'[[n], J]$ are zero vectors, and thus the $i_1, \dots, i_\alpha$-th elements of $T'[[n], J](\bar{x}_1, \dots, \bar{x}_{n-\alpha})$ are always zero. Therefore, if let $\bar{F}$ be a quadratic map obtained by substituting 0 into $x_{i_1}, \dots, x_{i_\alpha}$, then

$$\bar{P}(x_{j_1}, \dots, x_{j_{n-\alpha}}) = \bar{F} \circ T'[[n], J](\bar{x}_1, \dots, \bar{x}_{n-\alpha}).$$

From the form of equation (2), $\bar{F}$ can be seen as the central map of UOV in $n - \alpha$ variables. Therefore, we can regard $\bar{P}$ as the public key system of UOV in $n - \alpha$ variables.

Algorithm 2 describes the detail of this step.

Algorithm 2 Reduction to smaller UOV

```
$T' \leftarrow I_n$
$K \leftarrow \phi$
for $i = 1, \dots, |T_{row}|$ do
    $(a_1, \dots, a_n) \leftarrow T_{row}[i] \cdot T'$
    $k \leftarrow \{k' \mid a_{k'} \neq 0, k' \notin K\}$
    $T'' \leftarrow I_n$
    $(T'')_{k,j} \leftarrow -a_j/a_k \quad (j \notin K)$
    $T' \leftarrow T' \cdot T''$
    $K \leftarrow K \cup k$
end for
Substitue 0 to $\{x_k \mid k \in K\}$ of $P \circ T'$
```

3.3 Complexity

In this subsection, we describe the complexity of recovering the remaining part of T. We assume that the $i_1, \dots, i_\alpha$-th row vectors are recovered in Subsection 3.1, and $1 \leq i_1 < \cdots < i_{v'} \leq v < i_{v'+1} < \cdots < i_{v'+m'} \leq n$ $(v' + m' = \alpha)$.

In the resulting small system, every key recovery attack on UOV can be executed with a smaller complexity. However, it should be noted that we cannot obtain the number of vinegar and oil variables in the resulting system. Because this fact does not affect the complexity of the UOV attack [8], we chose the UOV attack to estimate the complexity of our fault attack. As described in Subsection 2.3, the complexity of UOV attack on the plain UOV is estimated to be $O\left(q^{v-m-1} \cdot m^4\right)$. Therefore, the complexity of the UOV attack on $\bar{P}$ is estimated as

$$O\left(q^{(v-m)-(v'-m')-1} \cdot m'^4\right). \tag{5}$$

If faults are caused randomly in the proposed attack, then v' is larger than m' with a high probability due to the form of equation (2). Hence, the proposed attack reduces the complexity of the UOV attack with a high probability.

In the proposed attack, the processes described in Subsections 3.1 and 3.2 can be explicitly executed in polynomial time. Therefore, in many cases, the complexity of equation (5) is dominant.

Remark 1 In this remark, we consider executing the proposed attack on the secret key T that is limited to a specific compact form.
In UOV, the secret linear map T can be restricted to a special form:

Table 1 The probability that the proposed attack reduces the complexity of recovering the secret key to each bit range

parameter	bit complexity				
	~ 40	$40 \sim 60$	$60 \sim 80$	$80 \sim 100$	$100 \sim$
UOV$(16, 60, 39)$	24.6%	18.7%	22.6%	22.8%	11.3%
UOV$(256, 50, 33)$	19.6%	10.9%	9.0%	15.6%	44.9%

$$T = \begin{pmatrix} I_{v\times v} & T' \\ 0_{m\times v} & I_{m\times m} \end{pmatrix}, \tag{6}$$

where T' is a $v \times m$ matrix. This limitation does not change the distribution of the public key map.

We suppose executing the proposed attack on UOV with a restricted T. For each row vector recovered in Subsection 3.1, we can identify which row vectors of T in the form of (6) the recovered vector corresponds to. If the recovered row vectors include some rows in the last m rows (they are unit vectors), they are dismissed because they are already obtained.

We assume that v' row vectors in the first v rows of T are recovered. Then, by using the processes described in Subsection 3.2, the UOV attack can be executed with

$$O\left(q^{(v-v')-m-1} \cdot m^4\right),$$

and this complexity is smaller than that of the proposed attack on UOV with a plain T.

Furthermore, in this case, it is clear that the reconciliation attack [4] can be executed more effectively. If $v - v' < m$, then solving a quadratic system of m equations in $v - v'$ variables is dominant in the reconciliation attack, and this is more effective than the existing attacks on UOV.

4. Theoretical Analysis

In this subsection, we theoretically analyze the complexity of our proposed attack on two parameter sets of UOV satisfying a 100-bit security. Two parameter sets $(q, v, m) = (16, 60, 39)$ and $(q, v, m) = (256, 50, 33)$ are chosen such that the complexities of the direct attack, UOV attack [8], reconciliation attack [4], and intersection attack [1] exceed 100 bits. The complexity of the proposed attack is estimated using equation (5).

Table 1 shows the probability that the proposed attack reduces the complexity of recovering the secret key to each bit range. For example, for UOV$(16, 60, 39)$, the complexity of the UOV attack applied during the proposed attack is lower than 40 bits with a 24.6% probability. These percentages are derived from 1000 simulations of the proposed attack.

As a result, the proposed attack executes the UOV attack on the resulting system with a smaller complexity than the claimed security level with approximately 90% probability on UOV$(16, 60, 39)$ and approximately 55% possibility on UOV$(256, 50, 33)$. The proposed attack is less effective on UOV$(256, 50, 33)$ because the complexity of the plain UOV attack on UOV$(256, 50, 33)$ is much larger than 100 bits.

5. Conclusion

In this paper, we propose a new fault attack on UOV, which is a multivariate signature scheme. This is the first fault attack on UOV that uses faults created on the secret key.

During the proposed attack, some row vectors of the linear map T of the secret key are recovered using faults generated on the central map F. Furthermore, we propose a new algebraic method for reducing a given public key to a smaller UOV system by using the partially recovered information of T. This reduction enables us to reduce the complexity of the UOV attack. As a result, for a parameter set UOV(16,60,39) satisfying 100-bit security, our simulation shows that the proposed attack recovers the secret key with a smaller complexity than the claimed security level with approximately 90% probability.

A naive countermeasure against the proposed attack is to check whether the secret key is faulty, and if so, to not generate the signature, as described in [6].

It should be noted that the proposed attack is also valid for Rainbow, a variant of UOV. However, we consider that the fault attack proposed in [6] is more effective than our proposed attack on Rainbow. Therefore, our future study will be to propose a more efficient fault attack on Rainbow.

Acknowledgments This work was supported by JST CREST Grant Number JPMJCR14D6 and JSPS KAKENHI Grant Number 21J20391.

References

[1] Beullens, W.: Improved cryptanalysis of UOV and Rainbow. IACR Cryptology ePrint Archive: Report 2020/1343 (2020)

[2] Beullens, W., Preneel, B.: Field lifting for smaller UOV public keys. In; INDOCRYPT 2017, LNCS, vol. 10698, pp. 227–246. Springer (2017)

[3] Ding, J., Schmidt, D.: Rainbow, a new multivariable polynomial signature scheme. In: ACNS 2005, LNCS, vol. 3531, pp. 164–175. Springer (2005)

[4] Ding, J., Yang, B., Chen, C.-O., Chen, M., Cheng, C.: New differential-algebraic attacks and reparametrization of Rainbow. In: ACNS 2008, LNCS, vol. 5037, pp. 242–257. Springer (2008)

[5] Garey, M.-R., Johnson, D.-S.: Computers and intractability: a guide to the theory of NP-completeness. W. H. Freeman (1979)

[6] Hashimoto, Y., Takagi, T., Sakurai, K.: General fault attacks on multivariate public key cryptosystems. In: PQCrypto 2011, LNCS, vol. 7071, pp. 1–18. Springer (2011)

[7] Kipnis, A., Patarin, J., Goubin, L.: Unbalanced oil and vinegar signature schemes. In: EUROCRYPT 1999, LNCS, vol. 1592, pp. 206–222. Springer (1999)

[8] Kipnis, A., Shamir, A.: Cryptanalysis of the oil and vinegar signature scheme. In: CRYPTO 1998, LNCS, vol. 1462, pp. 257–266. Springer (1998)

[9] Mus, K., Islam, S., Sunar, B.: QuantumHammer: a practical

hybrid attack on the LUOV signature scheme. In: CCS 2020, pp. 1071-1084. ACM (2020)

[10] NIST: Post-quantum cryptography CSRC. https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization

[11] NIST: Status report on the second round of the NIST post-quantum cryptography standardization process. NIST Internal Report 8309, NIST (2020)

[12] Shor, P. W.: Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. SIAM Journal on Computing **26**(5), pp. 1484–1509 (1997)