One vector to rule them all: Key recovery from one vector in UOV schemes

Pierre Pébereau
Sorbonne Université, LIP6, CNRS
Thales SIX

pierre.pebereau@lip6.fr

Unbalanced Oil and Vinegar is a multivariate signature scheme that was introduced in 1999. Most multivariate candidates for signature schemes at NIST's PQC standardization process are either based on UOV or closely related to it. The UOV trapdoor is a secret subspace, the "oil subspace". We show how to recover an equivalent secret key from the knowledge of a single vector in the oil subspace in any characteristic. The reconciliation attack was sped-up by adding some bilinear equations in the subsequent computations, and able to conclude after two vectors were found. We show here that these bilinear equations contain enough information to dismiss the quadratic equations and retrieve the secret subspace with linear algebra for practical parametrizations of UOV, in at most 15 seconds for modern instanciations of UOV.

This proves that the security of the UOV scheme lies in the complexity of finding exactly one vector in the oil space. In addition, we deduce a key recovery attack from any forgery attack by applying a corollary of our main result.

We show how to extend this result to schemes related to UOV, such as MAYO and VOX.

1 Introduction

In order to replace number-theoretic problems used in cryptography that are threatened by quantum computing, such as the discrete logarithm or factorization, several families of problems have been intensively studied. One of them is related to multivariate polynomial system solving, and is refered to as "multivariate cryptography". There are multiple arguments to support this direction. First of all, familiar (to the cryptographer) notions of algebraic geometry, studied in the case of elliptic curves for instance, can be reused in this context. Further, the underlying problem, Polynomial System Solving "PoSSo", is NP-hard, and this gives confidence in the hardness of this problem for quantum computers. Experimental data suggest that random instances are also hard. The history of this field shows that one must be very subtle in order to construct secure schemes with these tools, as in the past many cryptosystems that seemed secure turned out to be broken by

a wide variety of methods. Lately, the attacks on the Rainbow signature scheme have motivated a return to the roots, in particular to the parent scheme "Unbalanced Oil and Vinegar" of Kipnis, Patarin, Goubin [1].

The results are stated in as general form as possible, to allow more flexibility in their use. The idea behind this is that we believe such results could be applied to other collections of quadratic forms sharing a common totally isotropic subspace. An example of such a collection is the set of quadratic forms arising in the intersection attack of Beullens [2]. They naturally apply to relaxations of UOV, where the dimension of the secret subspace may be different from m.

Related work

Many contributions to the cryptanalysis of UOV stem from the study of Rainbow [3], a more structured scheme built upon the foundations of UOV. In particular, the reconciliation attack [4] targeted Rainbow but is easily applied to UOV. This attack finds vectors in the secret subspace of the UOV trapdoor by exploiting their relationship with each other. More recently, Beullens introduced the intersection attack [2] which improves the first step of the attack (finding the first (two) vector(s)). Beullens describes this reconciliation process in more detail in [5]. In that paper, he mentions that once enough vectors of $\mathcal{O}$ are found, one can dismiss the quadratic equations and solve a linear system. Using his bound, this process requires finding $\lfloor \frac{n}{k} \rfloor$ vectors in $\mathcal{O}$ before being able to conclude, which is 2 for modern UOV instanciations. Another key recovery attack against UOV is the Kipnis-Shamir attack, which targets invariant subspaces of some linear functions related to the publick key. This attack is the one that motivated the "unbalanced" property of UOV.

Previous work

In the context of side-channel attacks, more precisely fault-injection attacks, Thomas Aulbach, Fabio Campos, Juliane Krämer, Simona Samardjiska and Marc Stöttinger recently released a paper with a similar result [6], which they attribute to a comment of Ward Beullens. Their result can be stated in the same manner, namely that one vector yields a polynomial time key recovery. There is a fundamental difference in the reasoning and in the complexity achieved however, as they follow the intuition of Ward Beullens' reconciliation attack as described in the MAYO paper [5]: he observes that one needs only two vectors of the secret subspace to conclude because they induce an overdetermined linear system which admits $\mathcal{O}$ as its only solution. They use an adapted Kipnis-Shamir attack to obtain a second vector from the first one to conclude with this observation. In our case, we focus on the geometric point of view instead of the algebraic one. We show that only a single vector is enough to characterize $\mathcal{O}$, without using the reconciliation modelling. Therefore we skip directly from one vector to the full key, without using a second vector as a stepping stone. We obtain very efficient algorithms in practice for all parameters of UOV, where their attack suffers from the cost of the reconciliation attack. The largest instance they attack using their tools takes

a total of 12 hours by including the Kipnis-Shamir and the reconciliation step, while our attack takes only 13 seconds on the same instance. We also introduce a method to decide wether a vector belongs to the secret subspace or not, which is not possible with the tools introduced by [6] without going through the entire attack. This test was the original target of my work, and yielded the key recovery attack as an unforseen consequence.

Contribution

In this paper, we prove that the difficulty of retrieving the UOV secret key is not only dominated by the complexity of finding the first vector, which was assumed in the state-of-the-art attacks, but that in fact the problem becomes polynomial given a single vector in the secret subspace. Therefore, the complexity of retrieving the UOV secret key is *exactly* the difficulty of finding a single vector in the secret subspace. In addition, we show how this yields a polynomial-time answer to the question "$\boldsymbol{x} \in \mathcal{O}$?" without the secret key, which may be of independant interest. In particular, this yields a key recovery attack from any forgery attack.

2 Preliminaries

Notations

Let $\mathbb{F}_q$ for q a power of a prime denote the finite field with q elements. If $q = p^m$ for p prime, we call p the characteristic of $\mathbb{F}_q$. Vectors are assumed to be column vectors and are noted as bold letters: $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{o}, \ldots$. Matrices are noted as capital letters, and transposition is written A^T. Given a field $\mathbb{F}$ and an integer n, we note $\mathbb{F}[x_1, \cdots, x_n]$ or $\mathbb{F}[\boldsymbol{x}]$ the polynomial ring of $\mathbb{F}$ in n indeterminates. The restriction of a function f to a set E will be noted as $f_{|E}$

Quadratic forms

Let f be a quadratic form over a vector space $\mathbb{F}_q^n$. A function $\mathcal{F} : \boldsymbol{x} \mapsto (f_1(\boldsymbol{x}), \cdots, f_m(\boldsymbol{x}))$ such that each f_i is a quadratic form is called a *quadratic map*. In fields of odd characteristic, the knowledge of a quadratic form f is characterized by its *polar form* $f^* := (\boldsymbol{x}, \boldsymbol{y}) \mapsto f(\boldsymbol{x} + \boldsymbol{y}) - f(\boldsymbol{x}) - f(\boldsymbol{y})$ which is a symmetric bilinear form. As such, it admits a symmetric matrix representation in $\mathbb{F}_q^{n \times n}$ that we identify with it, and with the original quadratic form. In other words, given f a quadratic form, there exists $M \in \mathbb{F}_q^{n \times n}$ such that for all $\boldsymbol{x} \in \mathbb{F}_q^n$, $f(\boldsymbol{x}) = f^*(\boldsymbol{x}, \boldsymbol{x}) = \boldsymbol{x}^T M \boldsymbol{x}$. In fields of even characteristic, there is no longer an equivalence with symmetric bilinear forms, as symmetric forms are also antisymmetric. Instead, we can represent quadratic forms using triangular matrices. Note that this is also true in fields of odd characteristic, but there is no reason to dismiss the additional properties of symmetric bilinear forms when they are avalaible. In particular, the zero quadratic form has many equivalent representations

(any antisymmetric matrix) with triangular representation, while it corresponds to the zero matrix with the symmetric representation.

We say that f has *rank* r if the matrix associated to f has rank r. A subspace $V \subset \mathbb{F}^n$ is *isotropic* for f if there exists $\boldsymbol{x} \in V$ such that $f(\boldsymbol{x}) = 0$, *totally isotropic* if for all $\boldsymbol{x} \in V, f(\boldsymbol{x}) = 0$, and *anisotropic* if for all $\boldsymbol{x} \in V, f(\boldsymbol{x}) \neq 0$. For an introduction to quadratic forms, we refer the reader to Serre's *A course in arithmetic* [7].

We note here a characteristation of totally isotropic subspaces that is useful to characterize the secret key of UOV:

Lemma 1. *The subspace $\mathcal{O}$ is a totally isotropic subspace of a quadratic form f if and only if for all $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{O}^2, f^*(\boldsymbol{x}, \boldsymbol{y}) = f^*(\boldsymbol{y}, \boldsymbol{x}) = 0$.*

Observe that the dimension of a totally isotropic subspace of a quadratic form of a certain rank is bounded:

Lemma 2. *Let f a quadratic form of rank n defined over a field $\mathbb{K}$. Let $\mathcal{O}$ a totally isotropic subspace of f. Then $\mathcal{O}$ has dimension less than or equal to $\lfloor \frac{n}{2} \rfloor$.*

Proof. By contradiction, assume that $\dim(\mathcal{O}) = r > \lfloor \frac{n}{2} \rfloor$. Let B a basis of $\mathcal{O}$, let $\hat{B}$ the completion of B into a basis of $\mathbb{K}^n$. Then the matrix representing f^* in basis $\hat{B}$ has a block of zeros of size $r \times r$ in the top left. Therefore its rank is less than n, which is a contradiction.

□

Cryptanalysis

Given a signature scheme instance $\Sigma = (\mathcal{S}, \mathcal{P})$ where $\mathcal{S}$ is the secret key and $\mathcal{P}$ is the public key, we define two goals of cryptanalysis:

- *Forgery*, which is achieved if an attacker can find a signature for *one* message in the message space of Σ.
- *Key recovery*, which is achieved if the attacker obtains an equivalent secret key $\mathcal{S}'$ enabling them to sign *any* message.

These notions can be refined to specify the tools and goals of the attacker, but this high-level description is enough for us.

Unbalanced Oil and Vinegar signatures

One of the oldest multivariate signature schemes was introduced by Patarin [8], and later generalised with Kipnis and Goubin [1], and remains standing after more than two decades. We formulate it in a more abstract manner than in the seminal paper, following the formalism of Beullens [2].

Definition 3 (Patarin, Goubin, Kipnis [1])**.** *A UOV instance is parametrized by the following parameters:*

- *m, the number of equations*
- *n, the number of variables*
- *q, the size of the finite field $\mathbb{F}_q$.*

The UOV public key is a set of m quadratic forms $G = (G_1, ..., G_m)$ of rank n over $\mathbb{F}_q$. The secret key is a totally isotropic subspace $\mathcal{O}$ of dimension m of the homogeneous component of degree two of each G_i.

This property is not generic for a family of quadratic forms, and the key generation will use a trick to introduce this structure. This trick was the original formulation of UOV in [1], and corresponds to a block of zeros of size m in the top left corner of the symmetric matrices representing the key in a secret basis. In particular, the secret key is a pair (A, F) where A is a linear change of variables (that characterizes $\mathcal{O}$) and F is a quadratic map where the variables $x_i, 1 \leq i \leq m$ appear linearily. We deduce the public key as $G = F \circ A$ by composing the secret quadratic map with the secret change of variables. Write $A^{-1} = [\boldsymbol{o}_1, ..., \boldsymbol{o}_m, \boldsymbol{v}_1, ..., \boldsymbol{v}_{n-m}]$ and observe that $\mathcal{O} = \mathsf{span}(\boldsymbol{o}_1, ..., \boldsymbol{o}_m)$. For $1 \leq i \leq m$, we call x_i an "oil" variable, and the remaining ones "vinegar" variables. We note $v = n - m$, the number of vinegar variables.

To sign a message $\mu \in \{0,1\}^*$ the signer solves the system: $F(\boldsymbol{x}) = \mathcal{H}(\mu) \in \mathbb{F}_q^m$ where $\mathcal{H}$ is a *cryptographic hash function.* This is a linear system in the oil variables, with m unknowns and equations after choosing random values for the vinegar variables. The verifier, given $\boldsymbol{y} = A^{-1}\boldsymbol{x}$ and μ, checks that $G(\boldsymbol{y}) = \mathcal{H}(\mu)$.

We introduce the *forgery variety* which is the set of signatures accepted for a given vector $\boldsymbol{z} \in \mathbb{F}_q^m$. In practice, we always sign $\boldsymbol{z} = \mathcal{H}(\mu)$, but nothing stops a forgery attack from targeting a specific vector instead of a specific message. What is forbidden by the definition of $\mathcal{H}$ is given $\boldsymbol{z}$, finding μ such that $\mathcal{H}(\mu) = \boldsymbol{z}$.

Definition 4 (Forgery variety)**.** *Let G a UOV public key. Let $\boldsymbol{z} \in \mathbb{F}_q^m$. We define the* forgery variety associated to $\boldsymbol{z}$ *as the set of signatures of the vector $\boldsymbol{z}$:*

$$V(\boldsymbol{z}) = \{\boldsymbol{x} \in \mathbb{F}_q^n, 1 \leq i \leq m, G_i(\boldsymbol{x}) = m_i\}$$

This variety has dimension $n - m$. Notice that $\mathcal{O} \subset V(\mathbf{0})$.

It is interesting to note that the distribution of UOV signatures is not uniform in this forgery variety (which is exactly the set of accepted signatures for a given message). The name is motivated by the goal of a forgery attack against a signature scheme.

We include as a reference the parameters chosen for UOV in recent submissions at NIST, and for VOX which is very closely related to UOV.

3 Retrieving the UOV private key from one secret vector

In this section, we will assume that $n \leq 3m$. This is the case for all recent instanciations of UOV, in particular the ones referred to in figures 1 and 2 of the previous section.

	NIST SL	n	m	$\mathbb{F}_q$	|pk| (bytes)	|sk| (bytes)	|cpk| (bytes)	|sig+salt| (bytes)
ov-Ip	1	112	44	$\mathbb{F}_{256}$	278 432	237 912	43 576	128
ov-Is	1	160	64	$\mathbb{F}_{16}$	412 160	348 720	66 576	96
ov-III	3	184	72	$\mathbb{F}_{256}$	1 225 440	1 044 336	189 232	200
ov-V	5	244	96	$\mathbb{F}_{256}$	2 869 440	2 436 720	446 992	260

Figure 1: UOV parameters in [9]

Variant	Security Level	q	o	v	t	c	|sig|	|cpk|	|sk|	|csk|
VOX-I	128	251	8	9	6	6	102 B	9,104 B	35,056 B	64 B
VOX-III	192	1021	10	11	7	7	184 B	30,351 B	111,297 B	64 B
VOX-V	256	4093	12	13	8	8	300 B	82,400 B	292,160 B	64 B

Figure 2: VOX parameters in [10]

At the end of the section, we will explain how we proceed for very unbalanced cases ($n > 3m$), and some reasons why very unbalanced instances of UOV are unlikely to be used in practice.

We assume that we have acquired a single vector $\boldsymbol{x} \in \mathcal{O}$, the secret subspace, and leverage this information to complete an equivalent key recovery attack in polynomial time. To summarize, the secret subspace is included in the kernel of each dual linear form $\boldsymbol{x}^T G_i$ by definition. The intersection of the m hyperplanes defined by these kernels is of dimension $n - m$, and still contains $\mathcal{O}$. Therefore, this intersection is a smaller subspace than the ambiant space $\mathbb{F}_q^n$ that still contains the secret subspace, and even small enough to entirely retrieve the secret subspace by considering the restriction of the public key quadratic forms to this subspace.

Before we start, we introduce the Kipnis-Shamir attack that justifies that the cases $n \leq 2m$ will be called "easy instances of UOV".

Lemma 5 (Kipnis-Shamir cryptanalysis of Oil and Vinegar [11], [1])**.** *Let G a UOV public key with parameters n, m, q. Then the following holds:*

- *If $n = 2m$, there exists a probabilistic algorithm performing a key recovery attack against G in time $O(n^\omega)$.*
- *If $n > 2m$, there exists a probabilistic algorithm performing a key recovery attack against G in time $O(q^{n-2m} n^\omega)$.*
- *If $n < 2m$, there exists a deterministic algorithm performing a key recovery attack against G in time $O(n^\omega)$.*

Proof. The first two cases are exactly the Kipnis-Shamir attack against OV [11] and the extension to the unbalanced case found in [1]. The last case comes from the observation that if $n < 2m$, then the existence of a m-dimensional totally isotropic subspace for a quadratic form constrains its rank by lemma 2. Therefore we retrieve the subspace $\mathcal{O}$ by computing the kernels of the quadratic forms of the public key. Since $\dim(\mathcal{O}) = m$ and

G is composed of m matrices, and each matrix has a kernel that is a random subspace of positive dimension included in $\mathcal{O}$, we can expect to find a basis of $\mathcal{O}$ from these kernels. □

Lemma 6. *Let $G = (G_1, \cdots, G_m)$ a homogeneous quadratic map of rank n represented by matrices. Let $\mathcal{O}$ a common totally isotropic subspace of $G_1, \cdots, G_m$. Let $\boldsymbol{x} \in \mathcal{O} \setminus \{\mathbf{0}\}$. Let $J(\boldsymbol{x}) = (\boldsymbol{x}^T G_1, \cdots, \boldsymbol{x}^T G_m)$. Then $\mathcal{O} \subset \mathsf{Ker}(J(\boldsymbol{x}))$ which is an $(n-m)$-dimensional subspace.*

Observe that $2J(\boldsymbol{x})$ is the Jacobian of G, if the characteristic is not 2. This justifies the notation J.

Proof. Let $g(\boldsymbol{z}) = \boldsymbol{z}^T G_i \boldsymbol{z}$ for some i. By lemma 1, for all $\boldsymbol{z} \in \mathcal{O}$, we have: $g(\boldsymbol{x}) = g(\boldsymbol{z}) = 0$ and $g^*(\boldsymbol{z}, \boldsymbol{x}) = 0$. In particular, this implies that the kernel of the linear form $g_{\boldsymbol{x}} = g^*(\boldsymbol{x}, .)$ contains $\mathcal{O}$. By hypothesis, all the quadratic forms are of rank n, therefore this linear form is non-zero. Since it is a non-zero linear form, its kernel is a hyperplane.

We have shown that for all $1 \leq i \leq m, \mathcal{O} \subset \ker(\boldsymbol{x}^T G_i)$. Therefore,

$$\mathcal{O} \subset \bigcap_{1 \leq i \leq m} \ker(\boldsymbol{x}^T G_i) = \ker(J(\boldsymbol{x}))$$

The intersection of m hyperplanes has dimension $n-m$, which yields the conclusion. □

This lemma is the key to our attack. We apply it to the formalism of UOV in the following theorem:

Theorem 7 (Key recovery from one vector)**.** *Let $G = (G_1, \cdots, G_m)$ a UOV public key. Let $\mathcal{O}$ the secret subspace of G. Let $\boldsymbol{x} \in \mathcal{O} \setminus \{\mathbf{0}\}$.*

There exists an algorithm taking as input $(G, \boldsymbol{x})$ that outputs in polynomial time a basis of $\mathcal{O}$. More precisely, Algorithm 3a performs this task and has complexity $O(kn^\omega)$, where $2 \leq \omega \leq 3$ is the exponent of matrix multiplication.

Proof. By lemma 6, $\mathcal{O} \subset K(\boldsymbol{x}) = \ker(J(G)(\boldsymbol{x}))$. We compute the kernel $K(\boldsymbol{x})$ in time $O(n^\omega)$. Let $B \in \mathbb{F}_q^{n \times (n-m)}$ a basis of $K(\boldsymbol{x})$. Then, we restrict the public key to $K(\boldsymbol{x})$:

$$G_{|K(\boldsymbol{x})} := \text{For } 1 \leq i \leq m, G_{i|K(\boldsymbol{x})} = B^T G_i B \tag{1}$$

Computing the restrictions takes time $O(kn^\omega)$. By definition, $G_{|K(\boldsymbol{x})}$ is a UOV instance for parameters $(n-m, m)$. By hypothesis, $n \leq 3m$ therefore $n - m \leq 2m$. By lemma 5, such an instance is broken in time $O(n^\omega)$, yielding a basis of the subspace $\hat{\mathcal{O}}$, the secret subspace of $G_{|K(\boldsymbol{x})}$. Note that in practice, $n < 3m$ therefore we use the kernel approach instead of the Kipnis-Shamir attack. Once we obtain $\hat{\mathcal{O}}$, we take it back to the initial space $\mathbb{F}_q^n$ using B:

Let $C \in \mathbb{F}_q^{(n-m) \times m}$ a basis of $\hat{\mathcal{O}}$. Then, for all $g \in G$:

$$(B \cdot C)^T g (B \cdot C) = C^T (B^T g B) C = C^T g_{|K(\boldsymbol{x})} C = 0 \in \mathbb{F}_q^{m \times m}$$

This proves that $B \cdot C$ is a basis of $\mathcal{O}$ since it is a free family of maximal cardinality included in $\mathcal{O}$. This matrix product costs $O(n^\omega)$, which yields a total complexity $O(kn^\omega)$. □

```
one_vector_to_key(G, x)
 1 : m = |G|
 2 : K(x) = [x^T G_i   for 1 ≤ i ≤ m]
 3 : B = ker(K(x))
 4 : Ĝ = [B^T G_i B   for 1 ≤ i ≤ m]
 5 : C = []
 6 : for 1 ≤ i ≤ m :
 7 :       for z ∈ ker(Ĝ_i) :
 8 :           if z ∉ span(C) :
 9 :               C = C ∪ {z}
10 :       if |C| = m :
11 :           break
12 : return BC
```

(a) Key recovery from one vector

```
in_secret_subspace(G, x)
1 : n = |x|
2 : m = |G|
3 : K(x) = [x^T G_i   for 1 ≤ i ≤ m]
4 : B = ker(K(x))
5 : Ĝ = [B^T G_i B   for 1 ≤ i ≤ m]
6 : for 1 ≤ i ≤ m :
7 :       if rank(Ĝ_i) > 2(n − 2m) :
8 :           return false
9 : return true
```

(b) Fast test of $\boldsymbol{x} \in \mathcal{O}$?

Figure 3: Algorithms

Note that this result does not exploit the full structure of the problem in characteristic two since the bilinear forms associated to the matrix representing the quadratic forms are no longer symmetric. We obtain two linear forms per quadratic form with a single vector instead, which improves this result in even characteristic. We state it in the more general way because it is enough to be relevant for practical instances of UOV in even characteristic.

Notice that in the algorithm, we include a break statement because with overwhelming probability, if the kernels of the restrictions have dimension greater than one, a subset of them suffices to retrieve the public key. We also obtain the following result as a corollary of this theorem, which was the initial motivation for this work.

Corollary 8. *Given G a UOV public key and $x \in \mathbb{F}_q^n$, there exists a polynomial-time algorithm deciding wether $\boldsymbol{x} \in \mathcal{O}$.*

Note that this question is interesting only if $\boldsymbol{x}$ is in the forgery variety of the vector $\mathbf{0} \in \mathbb{F}_q^m$, as any vector that does not vanish the public key has no chance of being part of the secret subspace.

Intuitively, to prove the corollary, it suffices to apply the algorithm of Theorem 7 and conclude from a success or a failure. We do not need to apply the entirety of the algorithm, as we distinguish the case $\boldsymbol{x} \in \mathcal{O}$ using the rank of the restrictions of the

public key to $J(\boldsymbol{x})$. We use the following lemma to specialize the algorithm of Theorem 7 for this task.

Lemma 9. *Let G a collection of quadratic forms. Let $\boldsymbol{x} \in V(\mathbf{0})$, $\boldsymbol{x} \neq 0$. Let $J(\boldsymbol{x}) = (\boldsymbol{x}^T G_1, \cdots, \boldsymbol{x}^T G_m)$. Let B a basis of $\ker(J(\boldsymbol{x}))$. Then for all $g \in G$, $B^T g B$ has rank at most $n - m - 1$.*

Proof. Note that $\boldsymbol{x} \in \ker(J(\boldsymbol{x}))$: $J(\boldsymbol{x})\boldsymbol{x} = (\boldsymbol{x}^T G_1 \boldsymbol{x}, \ldots, \boldsymbol{x}^T G_m \boldsymbol{x}) = 0$ Therefore there exists $\lambda_1, ..., \lambda_{n-m}$ not all zero such that $\boldsymbol{x} = \sum_{i=1}^{n-m} \lambda_i B_i$. Let $\boldsymbol{x}' = (\lambda_1, \ldots, \lambda_{n-m})$. Then $\boldsymbol{x}'^T B^T G_i B = \boldsymbol{x}^T (G_i B) = (\boldsymbol{x}^T G_i) B$ and by definition B is a basis of $\ker(\boldsymbol{x}^T G_i)$ therefore $\boldsymbol{x}' \in \ker(B^T G_i B)$, which yields the upper bound on the rank of the G_i. □

Proof of Corollary 8. Let $J(\boldsymbol{x}) = (\boldsymbol{x}^T G_1, \cdots, \boldsymbol{x}^T G_m)$. Let B a basis of $\ker(J(\boldsymbol{x}))$. For all i, the rank of $B^T G_i B$ is upper bounded by lemma 9 since $\mathcal{O} \subset V(\mathbf{0})$. We show that if $\boldsymbol{x}$ is not in a larger linear subspace of $V(\mathbf{0})$ than $\mathsf{span}(x)$, then the kernel of the restrictions do not intersect on a larger subspace than $\mathsf{span}(x)$. Assume by contradiction that $\cap_{i=1}^{m} \ker(B^T G_i B)$ has dimension at least 2. Let $\boldsymbol{x}', \boldsymbol{y}'$ a basis of this subspace. Then define $\boldsymbol{x} := B\boldsymbol{x}'$ and $\boldsymbol{y} := B\boldsymbol{y}'$. Observe that for all i, $\boldsymbol{x}^T G_i \boldsymbol{y} = \boldsymbol{x}'^T B^T G_i B \boldsymbol{y}' = 0 = \boldsymbol{y}^T G_i \boldsymbol{x}$. But $\boldsymbol{x}, \boldsymbol{y}$ must be linearily independant since $\boldsymbol{x}', \boldsymbol{y}'$ were and B is a free family by definition. Therefore there is a dimension two totally isotropic subspace $\mathsf{span}(\boldsymbol{x}, \boldsymbol{y})$ shared by the G_i by Lemma 1, which is a contradiction.

Next, we claim that if $\boldsymbol{x} \in \mathcal{O}$, then the kernel of the G_i must be of a larger dimension and included in $\mathcal{O}$. To prove this, assume that $B = B_1 \oplus B_2$ where B_1 is a basis of $\mathcal{O}$ which is possible since $\mathcal{O} \subset \mathsf{span}(B)$. In this case, for all i,

$$B^T G_i B = \begin{pmatrix} 0 & C_1^{(i)} \\ C_1^{(i)T} & C_2^{(i)} \end{pmatrix}$$

where $C_1^{(i)} \in \mathbb{F}_q^{m \times (n-2m)}, C_2^{(i)} \in \mathbb{F}_q^{(n-2m) \times (n-2m)}$ and C_2 is symmetric. Due to the size of the block of zeros, such a matrix has rank at most $n - 2m + n - 2m = 2(n - 2m)$.

In general, we distinguish a vector of $\mathcal{O}$ from a generic vector of $V(\mathbf{0})$ if $2(n-2m) < n - m - 1 \iff n < 3m - 1$. If the parameters are such that $n = 3m$ or $3m - 1$, we can apply the general algorithm which will be a little slower but still polynomial. In practical instances of UOV, where $n = \frac{5}{2}m$, this rank is at most $2(n - 2m) = m$.

This yields algorithm 3b.

□

Very unbalanced instances of UOV

The attack described in the previous section only works if $n \leq 3m$ or if $n \leq 4m$ and q is even. We show here what happens in the opposite case. The algorithm of Theorem 7 does not yield an easy UOV instance, but instead a UOV instance that has some interesting properties.

Keeping with the formalism of Theorem 7, let $\hat{G} = G_{|K(\boldsymbol{x})}$ using the basis B of $K(\boldsymbol{x})$. This restriction can be defined regardless of the ratio $\frac{n}{m}$, and always corresponds to a

UOV instance in dimension $n-m$. Next, recall that $\boldsymbol{x} \in \mathsf{span}(B)$ and therefore we can define $\hat{\boldsymbol{x}} = (\lambda_1, \ldots, \lambda_{n-m})$ where $v = \sum_{i=1}^{n-m} \lambda_i B_i$. By construction, this vector $\hat{\boldsymbol{x}}$ is in the secret subspace of $\hat{G}$.

Notice both instances are equivalent since a solution of either can be translated to the other with the restriction basis B, and the restricted one is in dimension $n-m$ instead of n.

$$(G, \boldsymbol{x}, \mathcal{O}) \overset{B}{\leftrightarrow} (\hat{G}, \hat{\boldsymbol{x}}, \hat{O})$$

Further, by Lemma 9, this new UOV instance is composed of quadratic forms that are not full rank, and in particular which share a kernel included in $\mathcal{O}$. This information is redundant with the secret vector we had for the original instance, as this kernel corresponds to $\mathsf{span}(\hat{\boldsymbol{x}})$. We are tempted to use this new vector $\hat{\boldsymbol{x}}$ that belongs to $\hat{\mathcal{O}}$ to repeat the attack inductively, but this fails because this vector is in the kernel of each matrix of the public key, which means that the matrix $J(\boldsymbol{x})$ is the zero matrix. Therefore, we need to solve a new UOV instance (which has some more structure in the form of the kernel we observed in this paragraph) that is strictly weaker against key recovery attacks. For very unbalanced instances of UOV, we will need a constant number of vectors in the secret key to conclude, in a similar fashion as observed by Beullens in [5]. More precisely, each independant vector in $\mathcal{O}$ allows to reduce the search space by m dimensions. We can conclude with β vectors if $n - \beta m \leq 2m \iff \alpha - \beta \leq 2 \iff \beta \geq \alpha - 2 \iff \beta \geq \lceil \alpha - 2 \rceil$ since β is an integer. Naturally this yields $\beta = 1$ for practical instances of UOV.

There are two reasons why very unbalanced instances of UOV are unlikely to be used in practice:

1. It seems to be a bad idea to use very unbalanced UOV, because random polynomial systems are easier to solve when they are heavily unbalanced. An argument that justifies this statement is the generic algorithm of Thomae and Wolf (especially in characteristic two), and more generally the observation that any new variable is a degree of liberty that can be exploited for free.

2. UOV already has large keys. Linear increases in n yield quadratic increases in the key sizes.

This highlights an interesting compromise in the security of UOV: the larger the parameter $\alpha = \frac{n}{m}$, the stronger UOV is against key recovery attacks, and the weaker it is against forgery attacks. Reciprocally, the smaller α is, the weaker UOV is against key recovery attacks, and the stronger it is against forgery attacks.

Key recovery attack from any forgery attack

We have introduced an efficient algorithm testing wether a vector belongs to the secret subspace of a public key or not. This can be combined with any forgery attack to obtain a key recovery attack with the following observation: all the vectors of the secret subspace are valid signatures for the vector $\mathbf{0}$ for the homogeneized public key. Therefore, the

attacker repeatedly tries to obtain forgeries of the message $\mathbf{0}$ until he finds one that belongs to $\mathcal{O}$.

Note that it must be a forgery attack, and not a chosen message attack, as in practice the signer never signs a message, and instead signs a hash of the message. Therefore, querying a signature of the vector $\mathbf{0}$ implies finding pre-images of the vector $\mathbf{0}$ for the given hash function, which is assumed to be a hard task. This suggests that a signer should refuse to sign the vector $\mathbf{0}$ even in the hash and sign paradigm.

Experimental results

The algorithms we obtain have polynomial complexity. We show that they are also fast in practice by providing an implementation in sagemath [12], using native linear algebra functions. We test them against the parameter sets of [9], in even and odd characteristic (we use $q = 257$ in the odd case to have comparable field size.) The strategy is as follows: the oracle providing a vector in $\mathcal{O}$ is obtained by a function that chooses a random element in $\mathsf{span}(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_m)$, which are the first m columns of A^{-1}. The code can be found at :

`https://github.com/pi-r2/OneVector`

We test the attack against the parameters of [9], which are representative of the state-of-the-art instanciations of UOV. The hardware used is a laptop with an Intel CPU i7-1165G7 running at 2.80GHz with 8GB of RAM. All experiments were ran on a single thread.

n	112	160	184	244
Time	1.7s	4.4s	5.7s	13.3s

Figure 4: Key recovery from one vector with our attack in $\mathbb{F}_{256}$

To obtain a complete key recovery, one must first find a vector of the secret subspace $\mathcal{O}$. Then, the attacker uses the attack described in this paper to complete his basis of the secret subspace, in a matter of seconds on a laptop.

n	112	160	184	244
Time	0.2s	0.5s	0.7s	1.5s

Figure 5: "$\boldsymbol{x} \in \mathcal{O}$?" with our algorithm in $\mathbb{F}_{256}$

4 Application to UOV variants

MAYO

The MAYO signature scheme [5] was introduced by Ward Beullens as a generalization of UOV in which we allow the subspace $\mathcal{O}$ to have a smaller dimension than m. We switch to the notations of Beullens for clarity. o is the size of the secret subspace of a MAYO key, m is the number of quadratic forms in the public key, n remains the dimension of the vector space $\mathbb{F}_q^n$, and q will be a small power of two. In the UOV formalism used so far, $m = o$. In MAYO, o is significantly smaller than m. This relaxation makes the scheme much more compact, but increases signature size. Beullens introduces some additional structure in the form of a 'Whipping' transformation that maps $\mathbb{F}_q^n \to \mathbb{F}_q^{ko}$, instead of UOV which maps $\mathbb{F}_q^n \to \mathbb{F}_q^m$. This is required to allow the signer to sign. We obtain the UOV scheme for $m = 1$. We include below a set of parameters for MAYO as found in [5].

SL	no leakage	Parameters n	m	o	k	q	|pk| (Bytes)	|sig| (Bytes)
I	✗	66	67	5	14	16	518	494
	✓	67	68	6	14	16	730	501
III	✗	98	99	6	17	16	1055	881
	✓	99	102	6	20	16	1087	1038
V	✗	130	132	7	19	16	1864	1299
	✓	131	132	8	19	16	2392	1308

If we try to attack the UOV map of MAYO, then we consider a collection of m quadratic maps P_i, the public key maps, that share an o dimensional totally isotropic subspace. The attack proceeds as follows: Given $\boldsymbol{x} \in \mathcal{O}$, we obtain m linear forms $P_i'(\boldsymbol{x}, .)$, Therefore the intersection of their kernels generically defines $J(\boldsymbol{x})$ an $n - m$ dimensional subspace that still contains $\mathcal{O}$. In the context of MAYO, $n - m \leq o$. Therefore we recover $\mathcal{O}$ entirely from the kernels of the restriction of the public key to $J(\boldsymbol{x})$. Notice that this does not improve the reconciliation attack on MAYO, as this was already achieved by Beullens in [5] with the algebraic method. This shows that the work done in section 3 is coherent with the state of the art when transfered to MAYO.

VOX

To simplify the description, we will apply our attack to the more general formulation of VOX known as FOX. It is introduced in the same specification as VOX [10]. Notably, it relies on less assumptions than VOX and still has competitive signature sizes with UOV, with a priori improved security. This signature scheme is a UOV-like signature scheme where a constant number t of the secret key quadratic forms are random. This is known as the $\hat{+}$ perturbation. They are called 'vinegar forms' and the usual UOV quadratic forms are called 'oil forms' by analogy. The private key is then composed with

two changes of variables (S, T) where $S \in GL_o(\mathbb{F}_q), T \in GL_n(\mathbb{F}_q)$ In traditional UOV, $S = I_n$ and $T = A^{-1}$.

$$\mathcal{F} = S \circ \mathcal{P} \circ T$$

The transformation S adds "noise" to the equations: the oil quadratic forms are mixed with the vinegar quadratic forms. This implies that public system does not have a high dimensional totally isotropic subspace like the UOV one. More precisely, we have the following shape of S chosen in [10].

$$S = \begin{pmatrix} I_t & S' \\ 0 & I_{o-t} \end{pmatrix}, \quad S' \in \mathbb{F}_q^{(o-t)\times t} \tag{2}$$

The main takeaway is that S has $t(o-t)$ unknown coefficients. For vectors in $\mathcal{O}$, the contribution of the oil forms to these mixed equations is zero, therefore we can retrieve this linear change of variables with linear algebra from the evaluation of the public key on oil vectors. Each evaluation yields $o-t$ equations by expressing the last $o-t$ coefficients of $\mathcal{P}(\boldsymbol{x})$ as linear combinations of the first t coefficients. Therefore we need t vectors in the oil subspace to retrieve the change of variables S.

Once this is done, we can apply the tools introduced earlier to recover T from $\mathcal{P}' = S^{-1} \circ \mathcal{P}$ which is a UOV system with t random equations. If we are given $x \in \mathcal{O}$, we will observe that it only vanishes $m-t$ of the quadratic forms of $\mathcal{P}'$. Each of the remaining t vinegar forms have probability $\approx \frac{1}{q}$ to vanish coincidentally on this vector but the knowledge of S allows us to distinguish the oil forms. In any case, the algorithm $\boldsymbol{x} \in \mathcal{O}$? would enable one to distinguish oil forms from vinegar forms even if the equations were permuted.

Then, we will be able to reduce the $\mathcal{P}'$ instance to a smaller subspace of dimension $n-(m-t)$, as we will only consider $(m-t)$ linear forms instead of m. FOX with $S = I_n$, which is exactly what $\mathcal{P}'$ is, shares the weakness of UOV to the Kipnis-Shamir attacks (lemma 5), therefore we complete the attack if $n - m + t \leq 2m \iff n + t \leq 3m$. The parameters of FOX from [10] are in figure 6.

Variant	Security Level	q	o	v	t	|sig|	|cpk|	|csk|
FOX-I	128	251	48	72	8	120 B	47,056 B	64 B
FOX-III	192	4093	68	106	8	261 B	211,156 B	64 B
FOX-V	256	65521	91	140	8	462 B	694,892 B	64 B

Figure 6: FOX parameters in [10].

We have $n = o + v$, where $o = m$ in our formalism. In all cases $n \leq 2.55o$, and in particular $n + t = 122, 182, 239$ versus $3o = 144, 204, 273$. Therefore our attacks applies to these parameter sets of FOX, but only with knowledge of S, which we obtain from t vectors of $\mathcal{O}$.

It is interesting to note that the signer has to solve a random system involving t quadratic equations, therefore the scheme does not allow much flexibility in the choice of t, as this task can only be done quickly for small values of t.

5 References

[1] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar signature schemes. In Jacques Stern, editor, *Advances in Cryptology — EUROCRYPT '99*, pages 206–222, Berlin, Heidelberg, 1999. Springer Berlin Heidelberg.

[2] Ward Beullens. Improved cryptanalysis of uov and rainbow. In Anne Canteaut and François-Xavier Standaert, editors, *Advances in Cryptology – EUROCRYPT 2021*, pages 348–373, Cham, 2021. Springer International Publishing.

[3] Jintai Ding and Dieter Schmidt. Rainbow, a new multivariable polynomial signature scheme. In John Ioannidis, Angelos Keromytis, and Moti Yung, editors, *Applied Cryptography and Network Security*, pages 164–175, Berlin, Heidelberg, 2005. Springer Berlin Heidelberg.

[4] Jintai Ding, Bo-Yin Yang, Owen Chen, Ming-Shing Chen, and Doug Cheng. New differential-algebraic attacks and reparametrization of rainbow. Cryptology ePrint Archive, Paper 2008/108, 2008. `https://eprint.iacr.org/2008/108`.

[5] Ward Beullens. Mayo: Practical post-quantum signatures from oil-and-vinegar maps. In Riham AlTawy and Andreas Hülsing, editors, *Selected Areas in Cryptography*, pages 355–376, Cham, 2022. Springer International Publishing.

[6] Thomas Aulbach, Fabio Campos, Juliane Krämer, Simona Samardjiska, and Marc Stöttinger. Separating oil and vinegar with a single trace. Cryptology ePrint Archive, Paper 2023/335, 2023. `https://eprint.iacr.org/2023/335`.

[7] Jean-Pierre Serre. *A course in arithmetic*. Springer New York, NY, 1978.

[8] Jacques Patarin. The oil and vinegar signature scheme. In *Dagstuhl Workshop on Cryptography September, 1997*, 1997.

[9] Ward Beullens, Ming-Shing Chen, Shih-Hao Hung, Matthias Kannwischer, Bo-Yuan Peng, Cheng-Jhih Shih, and Bo-Yin Yang. Oil and vinegar: Modern parameters and implementations. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pages 321–365, 06 2023.

[10] Benoît Cogliati, Jean-Charles Faugère, Pierre-Alain Fouque, Louis Goubin, Robin Larrieu, Gilles Macario-Rat, Brice Minaud, and Jacques Patarin. Vox-sign, 2023.

[11] Aviad Kipnis and Adi Shamir. Cryptanalysis of the oil and vinegar signature scheme. In Hugo Krawczyk, editor, *Advances in Cryptology — CRYPTO '98*, pages 257–266, Berlin, Heidelberg, 1998. Springer Berlin Heidelberg.

[12] The Sage Developers. *SageMath, the Sage Mathematics Software System*, 2022. DOI 10.5281/zenodo.6259615.