Variables for Free: Fault Injection Attack on MAYO via Valid Solutions

Yadi Zhong

Auburn University, Auburn AL 36849, USA
yadi@auburn.edu

Abstract. Multivariate quadratic problem over a finite field, a NP-hard problem, is also considered as one of the hard problems for cryptanalytic-relevant quantum computers. It is the foundation of multivariate quadratic-based cryptography and several post-quantum digital signature schemes initially proposed in 1990s. Patarin's unbalanced Oil-and-Vinegar (UOV) scheme is the oldest MQ signature algorithm that remains secure against large-scale cryptanalytic-relevant quantum computers. UOV has compact signature size and succinct verification time. However, it suffers from a very large public key size. Subsequently, UOV-based variants have focused on minimizing the size of central map. MAYO, a candidate advanced to NIST's round 2 additional post-quantum digital signatures, is based off the UOV algorithm with whipped structure to reduce public key size. In this paper, we present a theoretical framework for fault injection-based attack. It targets the construction of valid signatures during the signing phase. This fault-based analysis allows efficient extraction of secret oil vectors from the public signatures, facilitating the complete recovery of the entire secret oil space $\mathcal{O}$. Finally, we show that this analysis is applicable to parameter sets of all three security level of all versions of MAYO.

1 Introduction

National Institute of Standards and Technology (NIST) finalized its first set of post-quantum cryptographic standards, FIPS 203, FIPS 204, FIPS 205 in August 2024. Both FIPS 203, a key encapulation mechanism (KEM), and FIPS 204, a digital signature, are built upon hard problems on lattice while FIPS 205 (SPHINCS$^+$) is a digital signature from secure hash functions. In March 2025, NIST additionally selected HQC, with underlying hard problem in decoding random quazi-cyclic codes, as the second KEM to be standardized. Regarding digital signatures selected by NIST for standardization, two out of three signatures, FIPS 204 and FN-DSA (to be published), i.e., Falcon, are based on lattices. In light of possible breaks of lattice-based cryptography of future development in lattice cryptanalysis, NIST is therefore interested in quantum-secure signature

This is the full version of conference paper [Zho25b].

alternatives with underlying hard problems other than the standardized lattice-based assumptions of FIPS 204 and FN-DSA. In 2022, NIST called for additional post-quantum signature schemes, where 40 schemes have been included in Round 1 in 2023. Last year, NIST announced 14 candidates advanced to the second round. Among these 14 signatures, four candidates are based on the hardness in solving multivariate quadratic systems – Unbalanced Oil-and-Vinegar (UOV) [BCD+24], and its variants, MAYO [BCC+24], SNOVA [WCD+24], and QR-UOV [AFI+24].

UOV, originally proposed by Kipnis, Patarin, and Goubin in 1999 [KPG99] after Kipnis and Shamir broke the balanced version of Oil and Vinegar [KS98], is one of the oldest, widely-studied multivariate signatures withstood attacks over 25+ years. It has the advantage of small signature size and fast verification compared with other post-quantum signature schemes. However, it suffers from large public and secret key sizes. In keeping the small signature and fast verification, UOV-based variants were proposed to further optimize key sizes. Among the common approaches are to add whipping structure (i.e., MAYO) and/or with extension field structure (i.e., QR-UOV and SNOVA). Since breaking of the third round finalist Rainbow [Beu22a,Beu21a], there have been a renewed interest in cryptanalysis of multivariate quadratic (MQ) problems. Recent results of XL-based attacks [FK21,FNT21], minrank attacks [FI23], wedge product attacks [Ran25,LBL+25], singular points [Péb24,Péb25], and partial key exposure attack [SFT25] have targeted UOV, MAYO, SNOVA, QR-UOV, and also MQ-Sign, a candidate from Korean post-quantum cryptography competition. Classical and quantum cryptanalysis evaluate security claims of post-quantum schemes against both traditional computing resources and large-scale cryptanalytic-relevant quantum computers. It inspires post-quantum constructions of threshold signatures and advanced primitives [CEN24,Zho25a], etc. Furthermore, side-channel attacks [ACK25,ACK+23,JD24b] as well as fault attacks [ACK25,FKNT22,JD24a,SMA+24] have focused on extracting secret key space on constraint devices and embedded systems implementing these MQ-based signatures, where adversaries additionally have access to the physical device.

1.1 Contribution

In this paper, we present the first theoretical fault attack framework on MAYO centered on its valid solutions during generation of a correct signature for any given message. Our proposed theoretical framework for fault attack against MAYO allows adversaries to efficiently extract valid oil vectors during the signing phase. With one or more secret oil vectors retrieved, the complete oil space $\mathcal{O}$, signer's secret, can be recovered under polynomial time when combining the latest results of Pébereau [Péb24]. Our contributions are the following:

– We began by highlighting the structure of MAYO, as a result of the emulsifiers proposed in MAYO. This vulnerability of non-unique solutions in solving the linear system of equations could be exploited by an adversary with

fault injection equipment to extract the secret oil vectors, which leads to the complete recovery of signer's secret key space.
- Stemmed from the structural properties of MAYO, we present a novel theoretical framework for fault injection attack. With our fault attack framework, MAYO's secret oil vector(s) and the entire oil space can be solved under polynomial time complexity. Our proposed fault attack is novel and it differs from all existing fault attacks against MAYO.
- We discuss applicability of our fault attack against MAYO parameter sets of all three security levels, as well as various versions of MAYO of Round 2 submission [BCC+24], Round 1 submission [BCC+23], ePrint [Beu21b], and SAC'21 [Beu22b]. From spatial point of view, we differentiate the changes made in MAYO signature parameter sets, and how it affects the proposed fault injection attack.

1.2 Outline

Section 2 covers the preliminaries in UOV and MAYO signature schemes. Section 3 presents our proposed theoretical framework and its corresponding fault injection attack against Round 2 submission of MAYO. Extension of the proposed attack to prior versions of MAYO signature schemes is described in Section 4. Finally, Section 5 concludes this work.

2 Preliminaries

2.1 Notions

We denote $\mathbb{F}_q$ as finite field of order q, where q is a prime number or a power of prime. We use $[a]$ the set of integers $\{1, \ldots, a\}$ for $a \in \mathbb{N}$. Vectors and matrices are in bold font. The ith row of matrix $\mathbf{A}$ as $\mathbf{A}[i,:]$, ith row and jth column as $\mathbf{A}[i,j]$ or $\mathbf{A}_{ij}$. The ith component of a vector $\mathbf{a}$ is written as $\mathbf{a}[i]$. Concatenation of two vectors $\mathbf{a}, \mathbf{b}$ is denoted as $\mathbf{a}||\mathbf{b}$. The upper triangular matrix $\mathsf{Upper}(\mathbf{A})$ of a square matrix $\mathbf{A}$ as $\mathsf{Upper}(\mathbf{A})_{ii} = \mathbf{A}_{ii}$ and $\mathsf{Upper}(\mathbf{A})_{ij} = \mathbf{A}_{ij} + \mathbf{A}_{ji}$ for $i < j$.

In this paper, we consider the homogeneous quadratic polynomial map $\mathcal{P}$, where no constant nor linear element exists in any quadratic polynomials in the map. Unlike UOV whose quadratic system of equations are computed under finite field $\mathbb{F}_q = \mathbb{F}_{256}$ with the exception of `ov-Is`, MAYO uses exclusively the extension field $\mathbb{F}_q = \mathbb{F}_{16}$ with irreducible polynomial $x^4 + x + 1$. For positive integers $m < n$, the multivariate quadratic map for UOV is an underdetermined systems of m quadratic equations under n input variables in $\mathbb{F}_{16}$, $\mathcal{P}(\mathbf{x}) = (p_1, \ldots, p_m) : \mathbb{F}_q^n \to \mathbb{F}_q^m$. Each quadratic polynomial p_i can be expressed in matrix multiplication with an upper-triangular matrix $\mathbf{P}_i$,

$$p_i(\mathbf{x}) = \mathbf{x}^\top \mathbf{P}_i \mathbf{x}.$$

Then, evaluating a multivariate quadratic map $\mathcal{P}$ under input vector $\mathbf{x}$ is done by computing the result of m quadratic polynomials $\{p_i(\mathbf{x})\}_{i \in [m]}$, where $\mathcal{P}(\mathbf{x}) = [p_1(\mathbf{x}), \ldots, p_m(\mathbf{x})] = \mathbf{y}$.

2.2 UOV

The conventional description of UOV by Patarin et. al. [KPG99] is centered at building the trapdoor function $\mathcal{P}$ from secret central map $\mathcal{F}$ and an invertible linear map $\mathcal{T}$. We instead adopt Beullens' description of UOV, where $\mathcal{P}$ vanishes on secret $\mathcal{O}$ space, as also consistent with the current UOV submission.

Definition 1 (Secret oil space $\mathcal{O}$). *Quadratic map $\mathcal{P}$ with UOV trapdoor vanishes on a linear subspace $\mathcal{O} \in \mathbb{F}_q^n$ with* $\dim(\mathcal{O}) = m$*,*

$$\mathcal{P}(\mathbf{o}) = 0, \quad \forall \mathbf{o} \in \mathcal{O}.$$

Definition 2 (Polar Form $\mathcal{P}'$). *The polar form of $\mathcal{P}$, or its differential map, is defined as $\mathcal{P}'$,*

$$\mathcal{P}'(x, y) = \mathcal{P}(x + y) - \mathcal{P}(x) - \mathcal{P}(y).$$

Note that $\mathcal{P}'$ has both symmetric and bilinear properties.

UOV signature scheme is described with the following algorithms, key generation, signing, and verification.

Key Generation. Secret key space $\mathcal{O}$ is spanned by row vectors of $(\mathbf{O}^\top \ \mathbf{I}_m)^\top$, where $\mathbf{O} \in \mathbb{F}_q^{(n-m)\times m}$. Matrix $\mathbf{O}$ can be deterministically expanded from a uniform seed $seed_{sk}$ with SHAKE256. Once $\mathbf{O}$ is obtained, the trapdoored multivariate quadratic map $\mathcal{P}$ is constructed such that all m multivariate quadratic polynomials vanish on $\mathcal{O}$. As multivariate quadratic polynomial $p_i(\mathbf{x}) = \mathbf{x}^\top \mathbf{P}_i \mathbf{x}$ is represented with upper triangular matrix $\mathbf{P}_i$, we can partition $\mathbf{P}_i$ as follows:

$$\mathbf{P}_i = \begin{pmatrix} \mathbf{P}_i^{(1)} & \mathbf{P}_i^{(2)} \\ 0 & \mathbf{P}_i^{(3)} \end{pmatrix}.$$

Then, for every $i \in [m]$, the following equation holds:

$$\left(\mathbf{O}^\top \ \mathbf{I}_m\right) \mathbf{P}_i \begin{pmatrix} \mathbf{O} \\ \mathbf{I}_m \end{pmatrix} = \mathbf{O}^\top \mathbf{P}_i^{(1)} \mathbf{O} + \mathbf{O}^\top \mathbf{P}_i^{(2)} + \mathbf{P}_i^{(3)} = \mathbf{0}$$

Submatrices $\mathbf{P}_i^{(1)} \in \mathbb{F}_q^{(n-m)\times(n-m)}$ (upper triangular) and $\mathbf{P}_i^{(2)} \in \mathbb{F}_q^{(n-m)\times m}$ are generated pseudorandomly from a uniformly random seed ($seed_{pk}$) with SHAKE256. Then, $\mathbf{P}_i^{(3)} \in \mathbb{F}_q^{m\times m}$ is derived directly from

$$\mathbf{P}_i^{(3)} = \mathsf{Upper}(-\mathbf{O}^\top \mathbf{P}_i^{(1)} \mathbf{O} - \mathbf{O}^\top \mathbf{P}_i^{(2)}),$$

with $\mathsf{Upper}(\cdot)$ transforms a square matrix to its upper-triangular form.

Signing. UOV signing process finds the preimage $\mathbf{s}$ such that its evaluation on quadratic map $\mathcal{P}(\mathbf{s}) = \mathbf{t}$ is satisfied, where $\mathbf{t}$ is hash digest of message M with random salt, $\mathbf{t} = \mathcal{H}(M||salt)$. As signer possesses secret trapdoor space $\mathcal{O}$, one can first randomly sample vinegar vector $\overline{\mathbf{v}} = (\mathbf{v}||\mathbf{0}_m) \in \mathbb{F}_q^n$ while setting the last m entries to zero. Then, the corresponding oil vector (if solution exists) is solved from

$$\mathcal{P}'(\overline{\mathbf{v}}, \mathbf{o}) = \mathbf{t} - \mathcal{P}(\overline{\mathbf{v}}) - \cancel{\mathcal{P}(\mathbf{o})} = \mathbf{t} - \mathcal{P}(\overline{\mathbf{v}}),$$

which is a system of linear equations instead. As secret oil vector $\mathbf{o}$ is the linear combination of m basis vectors in trapdoor space $\mathcal{O}$, it is written as $\mathbf{o} = (\mathbf{O}^\top \ \mathbf{I}_m)^\top \mathbf{x}$. Finding a solution of oil vector then becomes solving $\mathbf{x}$. The linear system of equations from differential map $\mathcal{P}'(\overline{\mathbf{v}}, \mathbf{o})$ is further simplified to

$$\mathbf{L}\mathbf{x} = \mathbf{t} - \mathcal{P}(\overline{\mathbf{v}}),$$

where matrices $\mathbf{L} \in \mathbb{F}_q^{m \times m}$ and $\mathbf{S}_i \in \mathbb{F}_q^{(n-m) \times m}$ are defined as

$$\mathbf{L}[i,:] = \mathbf{v}^\top \mathbf{S}_i \ ; \quad \left\{ \mathbf{S}_i = \left(\mathbf{P}_i^{(1)} + \mathbf{P}_i^{(1)\top} \right) \mathbf{O} + \mathbf{P}_i^{(2)} \right\}_{i \in [m]}$$

If the linear system is non-singular, it has a unique solution for vector $\mathbf{x}$. UOV signature $\mathbf{s}$ for digest $\mathbf{t} = \mathcal{H}(M||salt)$ is thus $\mathbf{s} = (\mathbf{v} + \mathbf{O}\mathbf{x})||\mathbf{x}$. If matrix $\mathbf{L}$ has $\mathsf{rank}(\mathbf{L}) < m$, signer restarts by choosing another vinegar vector and repeat the signing step.

Verification. The verifier receives message M, salt $salt$, signature $\mathbf{s}$, and locally computes $\mathcal{P}(\mathbf{s})$ and $\mathbf{t}' = \mathcal{H}(M||salt)$. Verifier accepts $\mathbf{s}$ as the valid signature if $\mathcal{P}(\mathbf{s}) = \mathbf{t}'$. Otherwise, signature $\mathbf{s}$ is rejected.

2.3 Structure of MAYO Signature

The main difference between MAYO and UOV is the dimension of secret oil space $\mathcal{O}$. UOV's $\mathcal{O}$ has a dimension of m, identical to the number of quadratic equations in the system $\mathcal{P}$, while MAYO has significantly downsized it to $\dim(\mathcal{O}) = o < m$. This technique allows MAYO to reduce its public key size. However, solving the quadratic system $\mathcal{P}(\mathbf{v}+\mathbf{o}) = \mathbf{t}$ of MAYO with its oil space becomes harder, where oil vector may not even exist to satisfy all constraints in system of m quadratic equation of map $\mathcal{P}$. Beullens [Beu21b] circumvent this issue by introducing a new structure – "whipping up" the public map $\mathcal{P}$ to a larger map $\mathcal{P}^*$ while accepting k input vectors $\mathbf{s} \in \mathbb{F}_q^n$,

$$\mathcal{P}^*(\mathbf{s}_1, \ldots, \mathbf{s}_k) = \sum_{i=1}^{k} \mathbf{E}_{ii} \mathcal{P}(\mathbf{s}_i) + \sum_{i=1}^{k} \sum_{j=i+1}^{k} \mathbf{E}_{ij} \mathcal{P}'(\mathbf{s}_i, \mathbf{s}_j),$$

$$\mathbf{E} = \begin{pmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} & \ldots & \mathbf{E}_{1k} \\ \mathbf{E}_{12} & \mathbf{E}_{22} & \ldots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{E}_{1k} & \ldots & \ldots & \mathbf{E}_{kk} \end{pmatrix},$$

where emulsifier matrices $\mathbf{E}_{ij} \in \mathbb{F}_q^{m \times m}$ are public parameters with property that all their non-trivial linear combinations have full rank. This is also called the multi-target whipped MQ problem. Multiplication by $\mathbf{E}_{ij}$ corresponds to multiplication by power of x modulo irreducible polynomials $f(x) \in \mathbb{F}_q[x]$ of degree m. For round 2 submission of MAYO, the parameter set included four irreducible polynomials in $\mathbb{F}_{16}[z]$:[1]

$$f_{64}(z) = z^{64} + x^3z^3 + xz^2 + x^3$$
$$f_{78}(z) = z^{78} + z^2 + z + x^3$$
$$f_{108}(z) = z^{108} + (x^2 + x + 1)z^3 + z^2 + x^3$$
$$f_{142}(z) = z^{142} + z^3 + x^3z^2 + x^2$$

Key Generation. MAYO's key generation is much similar to UOV's. A secret seed $seed_{sk}$ is generated randomly and expanded to derive the oil space $\mathcal{O}$ and public seed $seed_{pk}$. Both expanded secret and public keys esk, epk are derived from $seed_{sk}$ and $seed_{pk}$. Submatrices $\mathbf{P}_i^{(1)}, \mathbf{P}_i^{(2)}$ are determined pseudorandomly from $seed_{pk}$, and $\mathbf{P}_i^{(3)}$ is derived from $\mathbf{P}_i^{(3)} = \mathbf{Upper}(-\mathbf{O}^\top \mathbf{P}_i^{(1)} \mathbf{O} - \mathbf{O}^\top \mathbf{P}_i^{(2)})$.

Sign. MAYO follows UOV's signing step, where vinegar vector $\mathbf{v}$ is sampled randomly to solve for oil vector $\mathbf{o}$. From UOV, we have central map $\mathcal{P}$ vanishes on oil space $\mathcal{O}$. For MAYO, the extended oil space $\mathcal{O}^k$ is the kernel space of the whipped map $\mathcal{P}^*$. Details of sign algorithm is listed in Figures 1, 2 in Appendix.

Verification. Verifier check the validity of signature $sig = \mathsf{Encode}_{\mathsf{vec}}(s)||salt$ with the expanded public key epk, and message M. By computing

$$\mathbf{t} = \mathsf{SHAKE256}(\mathsf{SHAKE256}(M)||salt, \lceil m \log(q)/8 \rceil),$$

the verifier confirms if $\mathcal{P}^*(\mathbf{s}_1, \ldots, \mathbf{s}_k) = \mathbf{t}$. If not, signature is rejected. Details of verify algorithm is listed in Appendix, Figure 3.

3 Proposed Theoretical Framework for Fault Injection Attacks

The main difference of MAYO from UOV is the addition of whipped transformation with public emulsifier matrices. Initially proposed by Beullens, the hard problem underlying MAYO is based on the hardness of solving the multi-target whipped MQ problem, where a random MQ map $\mathcal{P}$ and its whipped up map $\mathcal{P}^*$ results in a multi-target preimage resistant function on average [BCC+24]. When UOV samples a random vinegar vector $\mathbf{v}$, it leads to the system of linear equations of

$$\mathbf{Lx} = \mathbf{t} - \mathcal{P}(\overline{\mathbf{v}}).$$

[1] MAYO has different set of m in its first round submission with irreducible polynomials of $f_{64}(z), f_{96}(z), f_{128}(z)$.

Note matrix $\mathbf{L}$ is square matrix. If $\mathbf{L}$ has full rank, then vector $\mathbf{x}$, the coefficients for the linear combination of secret oil vectors from space $\mathcal{O}$, is determined uniquely. However, notice the difference from the signing step of MAYO signature. Due to MAYO's whipped structure with whipping parameter k, when MAYO samples a random vinegar vector $\mathbf{v}$, the system of linear equations it constructed from Lines 15-27 of Figure 1 is

$$\mathbf{Ax} = \mathbf{y},$$

where matrix $\mathbf{A} \in \mathbb{F}_q^{m \times ko}$ is not a square matrix anymore. Whereas UOV requires $\mathbf{L}$ to be an invertible matrix for signature generation, such property cannot be said for the MAYO scheme. Based on the specification of MAYO, parameter sets of all three security levels have $ko > m$. We summarize as the following remark.

Remark 1 (Rank). In MAYO signature generation algorithm, the rank of the non-square matrix $\mathbf{A}$ representing the linear system of equations is $\mathsf{rank}(\mathbf{A}) \leq m$. The upper bound is $\mathsf{rank}(\mathbf{A}) = m$.

Note that it is possible that the rank of matrix $\mathbf{A}$ is smaller than m, with m being the upper bound. Remark 1 indicates that find a solution for the linear combination of the secret oil space $\mathcal{O}^k$, which satisfies $\mathbf{Ax} = \mathbf{y}$, is not unique. It is summarized in the following lemmas.

Lemma 1 (Free Variables). *For* $\mathsf{rank}(\mathbf{A}) = m$*, the number of free variables inside vector* $\mathbf{x}$ *for the linear system of equations* $\mathbf{Ax} = \mathbf{y}$ *is* $\mathsf{fv} = ko - m$*.*

Lemma 2 (Valid Signatures in MAYO sign). *Given a randomly sampled vinegar vector v, MAYO signature generation step produces* $\mathsf{vs} = \mathbf{q}^{\mathsf{fv}} = \mathbf{q}^{\mathbf{ko-m}}$ *valid signatures when its corresponding matrix* $\mathbf{A}$ *has rank of* $\mathsf{rank}(\mathbf{A}) = m$*.*

Based on both lemmas, when matrix $\mathbf{A}$ has rank of $\mathsf{rank}(\mathbf{A}) = m$, we have multiple valid solutions for finding $\mathbf{x}$ during the signing phase. Table 1 lists the number of free variables, and the total number of valid solutions for solving $\mathbf{Ax} = \mathbf{y}$ for all four parameter sets in Round 2 submission of MAYO, covering security levels I, III, and V.

If an adversary could exploit the deterministic version of MAYO's signing algorithm, it is possible to possess one or more oil vectors when applying both Lemmas 1 and 2. Beullens' reconciliation attack detailed in MAYO specification [BCC+24] also suggests a much lower complexity cost for finding the complete $\mathcal{O}$ space if one or more oil vectors are found. In addition, recent results by Pébereau [Péb24] showed that only polynomial time complexity is required for algorithms to successfully extract the entire oil space when the attacker possess a single oil vector. Based on works of [BCC+24,Péb24], we can build an efficient key recovery attack, summarized in Theorem 1.

Theorem 1. *Assume adversary* $\mathcal{A}$ *has access to the signing oracle, which allows* $\mathcal{A}$ *to rerun the signing algorithm with same message M and different vectors* $\mathbf{r}$ *of Line 14 of Figure 1. Under MAYO's deterministic signing algorithm, the*

probability of the adversary $\mathcal{A}$ performing a successful key recovery attack is lower-bounded by the probability of the sampled vinegar vector $\mathbf{v}$ *by* SHAKE256 *resulting in* $\mathsf{rank}(\mathbf{A}) = m$.

$$\mathsf{Pr}[\mathcal{A} = 1] \geq \mathsf{Pr}[\mathbf{v} \xleftarrow{\$} \mathsf{SHAKE256} : \mathsf{rank}(\mathbf{A}) = m]$$

Note that the restart probability for Round 2 MAYO is between 2^{-12} *and* 2^{-20} *[BCC+24].*

For adversary to have the capability of altering vector $\mathbf{r}$, he/she needs to have access to fault injection equipment, where faults are injected, i.e., laser-based faults or voltage-induced glitches, to physical device, i.e., micro-controllers or FPGAs, running the MAYO's signature signing algorithm.

3.1 Construction of Fault Injection Attack

Our attack centers around the theoretical framework of Section 3, where free variables in $\mathbf{Ax} = \mathbf{y}$ leads to many valid signatures under the same $\mathbf{A}$ matrix of $\mathsf{rank}(\mathbf{A}) = m$. As matrix $\mathbf{A}$ is non-square and non-invertible, the solution $\mathbf{x}$ for linear system of equations heavily depends on the numeric value of random vector $\mathbf{r}$ (derived from SHAKE256). In particular, it is reliant on the entries in $\mathbf{r}$ which correspond to the locations of free variables in $\mathbf{Ax} = \mathbf{y}$. Therefore, changes made in $\mathbf{r}$, or on these entries in $\mathbf{r}$, result in a new valid signature $\mathbf{s}'$ for the same message M. If an adversary is able to ask the signing algorithm to produce one or more valid signatures (beyond the original $\mathbf{s}$) for the identical message based on Section 3, then the attacker obtains recover one or more oil vectors, and in turn recovers the entire secret space based on the attack from Pébereau [Péb24].

To generate a different $\mathbf{r}$ other than what is intended from algorithm, we recourse to fault injection during run-time. Note that faults attack proposed in this paper is novel and it differs from all existing strategies such as creating faulty signatures, preventing the addition of oil and vinegar, injecting faults to oil space $\mathcal{O}$, flipping bits in secret matrices of esk, or disturbing linear system setup, or skipping the new assignment of $\mathbf{A}$, etc., as summarized by Aulbach et. al. [ACK25].

For our attack, faults are needed for producing different $\mathbf{r}$ vectors than the original one derived from SHAKE256. Signatures generated under faults are still valid signatures for any given messages. Thus, the proposed fault attack evades detection from fault-based countermeasures like validity checks. From theoretical perspective, faults can be injected during either stages in the signing algorithm. We describe them as below:

1. **When r is assigned**: Vector $\mathbf{r}$ is first instantiated on Line 14 of Figure 1. Faults can be injected to perturb the decode process from byte-string to vector of the SHAKE256 output V. This can be done with voltage glitches so that some memory cells may contain logic values which deviate from V. It is also possible to inject fault at the respective locations in memory (or stack, or registers/flip flops) after $\mathbf{r}$ is assigned. This would generally require additional knowledge of the memory organization of the physical device and the location where $\mathbf{r}$ is stored, along with high precision of fault injection.

2. **Inside SampleSolution() function call**: Vector $\mathbf{x}$ is constructed with values from $\mathbf{r}$ on Line 2 of subroutine SampleSolution() in Figure 2. Also, matrix multiplication $\mathbf{Ar}$ is computed in Line 3 in Figure 2. Faults can be injected into both places to alter the value $\mathbf{r}$.

Fault injection during any one of the above scenarios would possibly result in a different randomized values for the $\mathsf{fv} = ko - m$ free variables. At the end of the signing phase, the signature output should be valid for the message M and salt *salt*. Oil vector can then be derived by comparing valid signature pair(s). It is feasible to also obtain multiple valid signatures of the same message M and salt pair. Therefore, combined with the technique from Pébereau [Péb24], the adversary can recover the entire oil space in polynomial time when only possesses a single oil vector. Additional time will be saved if multiple oil vectors are extracted during the fault injection attack with multiple valid signatures.

4 Extension to the Complete Parameter Sets in All Versions of MAYO [BCC+24,BCC+23,Beu22b,Beu21b]

Round 2 submission of MAYO offers four parameter sets, MAYO_1 and MAYO_2 for security level I while MAYO_3 and MAYO_5 target security level III and V, respectively. From Table 1, MAYO_1 has 2 free variables and $16^2 = 256$ possible valid signatures for a given $\mathbf{A}$ with $\mathsf{rank}(\mathbf{A}) = m$. MAYO_2 has 4 free variables and $16^4 = 65,536$ possible valid signatures for a given $\mathbf{A}$ of $\mathsf{rank}(\mathbf{A}) = m$. For security level III and V, MAYO_3 and MAYO_5 each has 2 free variables and $16^2 = 256$ possible valid signatures, similar to MAYO_1.

In addition to the main parameter sets of MAYO in Table 1, the round 2 specification provides 29 additional parameters for possible trade-offs between public key size and signature size covering all three security levels, I, III, and V. Based on the analysis on Section 3, the number of free variables $\mathsf{fv} = ko - m$ and valid signatures $\mathsf{vs} = q^{\mathsf{fv}}$ are listed in Table 2. From Table 2, the parameter set of $\{n = 155, m = 144, o = 11, k = 14\}$ contains the most free variables of $\mathsf{fv} = ko - m = 10$ with $\mathsf{vs} = 16^{10}$ possible solutions for a signature when the linear system of equations has $\mathsf{rank}(\mathbf{A}) = m$. For security level I, parameter set of $\{n = 81, m = 70, o = 11, k = 7\}$ gives $\mathsf{fv} = ko - m = 7$ free variables and $\mathsf{vs} = 16^7$ valid signatures. Similarly, among parameters of security level III, parameter set of $\{n = 115, m = 100, o = 15, k = 7\}$ gives $\mathsf{fv} = ko - m = 5$ free variables and $\mathsf{vs} = 16^5$ valid signatures. Therefore, the proposed theoretical attack against MAYO is suitable for the additional parameters of MAYO as well.

For better comparison, we also list the number of free variables fv and valid solutions vs for Round 1 submission of MAYO's parameter sets in Table 3, and its additional parameters in Table 4. For security level V, parameter set of $\{n = 133, m = 128, o = 12, k = 12\}$ in Table 3 and parameter $\{n = 137, m = 128, o = 16, k = 9\}$ both have $\mathsf{fv} = ko - m = 16$ free variables and $\mathsf{vs} = 16^{16}$ valid signatures. Similarly, parameter sets of $\{n = 69, m = 64, o = 11, k = 10\}$ of security level I and $\{n = 100, m = 96, o = 12, k = 12\}$ in security level III

Table 1: Parameter set for Round 2 submission of MAYO. Note the number of free variables fv and total number of valid solutions vs for one sampled vinegar vector with $\mathsf{rank}(\mathbf{A}) = m$ for matrix $\mathbf{A}$.

Parameter set	**n**	**m**	**o**	**k**	$\mathsf{fv} = \mathbf{ko} - \mathbf{m}$	$\mathsf{vs} = \mathbf{q}^{\mathbf{ko}-\mathbf{m}}$
MAYO_1	86	78	8	10	2	16^2
MAYO_2	81	64	17	4	4	16^4
MAYO_3	118	108	10	11	2	16^2
MAYO_5	154	142	12	12	2	16^2

Table 2: Additional parameter set for Round 2 submission of MAYO with security levels I, III, V. Free variables fv and valid solutions vs are computed.

Security Level	**n**	**m**	**o**	**k**	$\mathsf{fv} = \mathbf{ko} - \mathbf{m}$	$\mathsf{vs} = \mathbf{q}^{\mathbf{ko}-\mathbf{m}}$
Level I	85	76	9	9	5	16^5
	84	74	10	8	6	16^6
	81	70	11	7	**7**	$\mathbf{16^7}$
	80	68	12	6	4	16^4
	80	66	14	5	4	16^4
	86	64	22	3	2	16^2
	100	64	33	2	2	16^2
Level III	123	114	9	13	3	16^3
	117	106	11	10	4	16^4
	116	104	12	9	4	16^4
	115	102	13	8	2	16^2
	115	100	15	7	**5**	$\mathbf{16^5}$
	115	98	17	6	4	16^4
	118	98	20	5	2	16^2
	121	96	25	4	4	16^4
	129	96	33	3	3	16^3
	150	96	49	2	2	16^2
Level V	156	146	10	15	4	16^4
	155	144	11	14	**10**	$\mathbf{16^{10}}$
	153	140	13	11	3	16^3
	152	138	14	10	2	16^2
	152	136	16	9	8	16^8
	151	134	17	8	2	16^2
	152	132	20	7	8	16^8
	155	132	23	6	6	16^6
	157	130	27	5	5	16^5
	161	128	33	4	4	16^4
	172	128	44	3	4	16^4
	202	128	65	2	2	16^2

Table 3: Parameter set for Round 1 submission of MAYO. Free variables fv and valid solutions vs are listed.

Parameter set	**n**	**m**	**o**	**k**	$\mathsf{fv} = \mathbf{ko} - \mathbf{m}$	$\mathsf{vs} = \mathbf{q}^{\mathbf{ko}-\mathbf{m}}$
MAYO_1	66	64	8	9	8	16^8
MAYO_2	78	64	18	4	8	16^8
MAYO_3	99	96	10	11	14	16^{12}
MAYO_5	133	128	12	12	16	16^{14}

Table 4: Additional parameter set for Round 1 submission of MAYO with security levels I, III, V. Free variables fv and valid solutions vs are listed.

Security Level	**n**	**m**	**o**	**k**	$\mathsf{fv} = \mathbf{ko} - \mathbf{m}$	$\mathsf{vs} = \mathbf{q}^{\mathbf{ko}-\mathbf{m}}$
Level I	66	65	7	11	12	16^{12}
	67	64	9	8	8	16^8
	69	64	11	7	**13**	$\mathbf{16^{13}}$
	70	64	12	6	8	16^8
	73	64	15	5	11	16^{11}
	76	64	18	4	8	16^8
	82	64	24	3	8	16^8
	106	64	36	2	8	16^8
Level III	99	97	9	12	11	16^{11}
	100	96	11	10	**14**	$\mathbf{16^{14}}$
	101	96	12	9	12	16^{12}
	102	96	13	8	8	16^8
	104	96	15	7	9	16^9
	107	96	18	6	12	16^{12}
	110	96	21	5	9	16^9
	115	96	26	4	8	16^8
	124	96	35	3	9	16^9
	156	96	52	2	8	16^8
Level V	131	130	10	14	10	16^{10}
	132	129	11	13	14	16^{14}
	134	128	13	11	15	16^{15}
	135	128	14	10	12	16^{12}
	137	128	16	9	**16**	$\mathbf{16^{16}}$
	138	128	17	8	8	16^8
	141	128	20	7	12	16^{12}
	144	128	23	6	10	16^{10}
	149	128	28	5	12	16^{12}
	155	128	34	4	8	16^8
	167	128	46	3	10	16^{10}
	208	128	68	2	8	16^8

Table 5: Parameter sets for ePrint manuscript (updated in 2022) of MAYO signature scheme [Beu21b], including parameters with and without leakage of secret key. Free variables fv and valid solutions vs are computed.

Security Level	**No Leakage**	**n**	**m**	**o**	**k**	$\mathsf{fv} = \mathbf{ko - m}$	$\mathsf{vs} = \mathbf{q}^{\mathbf{ko-m}}$
Level I	✗	66	67	5	14	3	16^3
	✓	67	68	6	14	**16**	$\mathbf{16^{16}}$
Level III	✗	98	99	6	17	3	16^3
	✓	99	102	6	20	**18**	$\mathbf{16^{18}}$
Level V	✗	130	132	7	19	1	16^1
	✓	131	132	8	19	**20**	$\mathbf{16^{20}}$

of the additional parameters each allow $\mathsf{fv} = 13$ and $\mathsf{fv} = 14$ free variables and $\mathsf{vs} = 16^{13}$ and $\mathsf{vs} = 16^{14}$ valid signatures. To summarize, the Round 1 MAYO parameters on average produces more free variables and valid solutions if using the deterministic signing scheme with the fixed message M.

Table 6: Parameter sets for initial publication of MAYO signature scheme in SAC'21 [Beu22b]. Free variables fv and valid solutions vs are listed. Finite field $\mathbb{F}_q = \mathbb{F}_7$.

Security Level	**No Leakage**	**n**	**m**	**o**	**k**	$\mathsf{fv} = \mathbf{ko - m}$	$\mathsf{vs} = \mathbf{q}^{\mathbf{ko-m}}$
Level I	✗	74	76	6	13	2	7^2
	✓	76	78	7	15	**33**	$\mathbf{7^{33}}$
Level III	✗	111	117	6	20	3	7^3
	✓	112	117	7	20	**23**	$\mathbf{7^{23}}$
Level V	✗	148	152	8	20	8	7^8
	✓	149	157	7	26	**25**	$\mathbf{7^{25}}$

Finally, we provide the statistics for MAYO parameters of its ePrint manuscript (updated in 2022) [Beu21b] in Tables 6 and parameter sets in its initial publication in SAC'21 [Beu22b] in Table 5. For parameters with no leakage of secret keys identified by Beullens, they have much higher values for free variables fv and valid solutions vs than those in MAYO's Round 1 and 2 parameter sets.

5 Conclusion and Countermeasure

In conclusion, we presented a novel theoretical attack framework against MAYO that targets the construction of valid signatures during the signing phase. This provides the adversary with an efficient extraction of secret oil vectors from

public signatures, which further allows the complete recovery of the entire secret oil space $\mathcal{O}$ when combined with the recent results from Pébereau [Péb24]. We then showed that this attack is applicable to all parameter sets of the round 2 submission of MAYO. To prevent the attack proposed in this paper, we recommend (*i*) using the randomized version of the signing algorithm instead of the deterministic one; (*ii*) minimizing the number of free variables $\mathsf{fv} \to 0$. This will mitigate any adversarial strategy of getting the identical salt value when running the signature generation twice, or multiple time with the same message. Lastly, we think the findings in this paper may be helpful for future work analyzing the forgery vulnerabilities in MAYO signatures.

Acknowledgements Author would like to thank the MAYO team for their helpful comments.

References

[ACK+23] Thomas Aulbach, Fabio Campos, Juliane Krämer, Simona Samardjiska, and Marc Stöttinger. Separating oil and vinegar with a single trace side-channel assisted kipnis-shamir attack on UOV. *IACR TCHES*, 2023(3):221–245, 2023.

[ACK25] Thomas Aulbach, Fabio Campos, and Juliane Krämer. SoK: On the physical security of UOV-based signature schemes. In Ruben Niederhagen and Markku-Juhani O. Saarinen, editors, *Post-Quantum Cryptography - 16th International Workshop, PQCrypto 2025, Part I*, pages 199–231, Taipei, Taiwan, April 08–10, 2025. Springer, Cham, Switzerland.

[AFI+24] Rika Akiyama, Hiroki Furue, Yasuhiko Ikematsu, Fumitaka Hoshino, Koha Kinjo, Haruhisa Kosuge, Satoshi Nakamura, Shingo Orihara, Tsuyoshi Takagi, and Kimihiro Yamakoshi. QR-UOV. Technical report, National Institute of Standards and Technology, 2024. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

[BCC+23] Ward Beullens, Fabio Campos, Sofía Celi, Basil Hess, and Matthias J. Kannwischer. MAYO. Technical report, National Institute of Standards and Technology, 2023. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures.

[BCC+24] Ward Beullens, Fabio Campos, Sofía Celi, Basil Hess, and Matthias J. Kannwischer. MAYO. Technical report, National Institute of Standards and Technology, 2024. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

[BCD+24] Ward Beullens, Ming-Shing Chen, Jintai Ding, Boru Gong, Matthias J. Kannwischer, Jacques Patarin, Bo-Yuan Peng, Dieter Schmidt, Cheng-Jhih Shih, Chengdong Tao, and Bo-Yin Yang. UOV — Unbalanced Oil and Vinegar. Technical report, National Institute of Standards and Technology, 2024. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

[Beu21a] Ward Beullens. Improved cryptanalysis of UOV and Rainbow. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part I*, volume 12696 of *LNCS*, pages 348–373, Zagreb, Croatia, October 17–21, 2021. Springer, Cham, Switzerland.

[Beu21b] Ward Beullens. MAYO: Practical post-quantum signatures from oil-and-vinegar maps. Cryptology ePrint Archive, Report 2021/1144, 2021.

[Beu22a] Ward Beullens. Breaking rainbow takes a weekend on a laptop. In Yevgeniy Dodis and Thomas Shrimpton, editors, *CRYPTO 2022, Part II*, volume 13508 of *LNCS*, pages 464–479, Santa Barbara, CA, USA, August 15–18, 2022. Springer, Cham, Switzerland.

[Beu22b] Ward Beullens. MAYO: Practical post-quantum signatures from oil-and-vinegar maps. In Riham AlTawy and Andreas Hülsing, editors, *SAC 2021: 28th*, volume 13203 of *LNCS*, pages 355–376, Virtual Event, September 29 – October 1, 2022. Springer, Cham, Switzerland.

[CEN24] Sofia Celi, Daniel Escudero, and Guilhem Niot. Share the MAYO: thresholdizing MAYO. Cryptology ePrint Archive, Report 2024/1960, 2024.

[FI23] Hiroki Furue and Yasuhiko Ikematsu. A new security analysis against MAYO and QR-UOV using rectangular MinRank attack. In Junji Shikata and Hiroki Kuzuno, editors, *IWSEC 23*, volume 14128 of *LNCS*, pages 101–116, Yokohama, Japan, August 29–31, 2023. Springer, Cham, Switzerland.

[FK21] Hiroki Furue and Momonari Kudo. Polynomial XL: A variant of the XL algorithm using macaulay matrices over polynomial rings. Cryptology ePrint Archive, Report 2021/1609, 2021.

[FKNT22] Hiroki Furue, Yutaro Kiyomura, Tatsuya Nagasawa, and Tsuyoshi Takagi. A new fault attack on UOV multivariate signature scheme. In Jung Hee Cheon and Thomas Johansson, editors, *PQCrypto 2022*, pages 124–143, Virtual Event, September 28–30, 2022. Springer, Cham, Switzerland.

[FNT21] Hiroki Furue, Shuhei Nakamura, and Tsuyoshi Takagi. Improving Thomae-Wolf algorithm for solving underdetermined multivariate quadratic polynomial problem. In Jung Hee Cheon and Jean-Pierre Tillich, editors, *PQCrypto 2021*, pages 65–78, Daejeon, South Korea, July 20–22, 2021. Springer, Cham, Switzerland.

[JD24a] Sönke Jendral and Elena Dubrova. MAYO key recovery by fixing vinegar seeds. *CiC*, 1(4):17, 2024.

[JD24b] Sönke Jendral and Elena Dubrova. Single-trace side-channel attacks on MAYO exploiting leaky modular multiplication. Cryptology ePrint Archive, Report 2024/1850, 2024.

[KPG99] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced Oil and Vinegar signature schemes. In Jacques Stern, editor, *EUROCRYPT'99*, volume 1592 of *LNCS*, pages 206–222, Prague, Czech Republic, May 2–6, 1999. Springer Berlin Heidelberg, Germany.

[KS98] Aviad Kipnis and Adi Shamir. Cryptanalysis of the Oil & Vinegar signature scheme. In Hugo Krawczyk, editor, *CRYPTO'98*, volume 1462 of *LNCS*, pages 257–266, Santa Barbara, CA, USA, August 23–27, 1998. Springer Berlin Heidelberg, Germany.

[LBL+25] Thai Hung Le, Maxime Bros, Jacob Lichtinger, Brice Minaud, Ray Perlner, Daniel Smith-Tone, and Cristian Valenzuela. Exploiting SNOVA's Structure in the Wedge Product Attack. NIST 6th PQC Standardization Conference, 2025.

[Péb24] Pierre Pébereau. One vector to rule them all: Key recovery from one vector in UOV schemes. In Markku-Juhani Saarinen and Daniel Smith-Tone, editors, *PQCrypto 2024, Part II*, pages 92–108, Oxford, UK, June 12–14, 2024. Springer, Cham, Switzerland.

[Péb25] Pierre Pébereau. Singular points of UOV and VOX. In Serge Fehr and Pierre-Alain Fouque, editors, *EUROCRYPT 2025, Part VI*, volume 15606 of *LNCS*, pages 294–323, Madrid, Spain, May 4–8, 2025. Springer, Cham, Switzerland.

[Ran25] Lars Ran. Wedges, oil, and vinegar – an analysis of UOV in characteristic 2. Cryptology ePrint Archive, Report 2025/1143, 2025.

[SFT25] Yuki Seto, Hiroki Furue, and Atsushi Takayasu. Partial key exposure attacks on UOV and its variants. In Willy Susilo and Josef Pieprzyk, editors, *ACISP 25, Part II*, volume 15659 of *LNCS*, pages 104–123, Wollongong, NSW, Australia, July 14–16, 2025. Springer, Singapore, Singapore.

[SMA+24] Oussama Sayari, Soundes Marzougui, Thomas Aulbach, Juliane Krämer, and Jean-Pierre Seifert. HaMAYO: A fault-tolerant reconfigurable hardware implementation of the MAYO signature scheme. In Romain Wacquez and Naofumi Homma, editors, *COSADE 2024*, volume 14595 of *LNCS*, pages 240–259, Gardanne, France, April 9–10, 2024. Springer, Cham, Switzerland.

[WCD+24] Lih-Chung Wang, Chun-Yen Chou, Jintai Ding, Yen-Liang Kuan, Jan Adriaan Leegwater, Ming-Siou Li, Bo-Shu Tseng, Po-En Tseng, and Chia-Chun Wang. SNOVA. Technical report, National Institute of Standards and Technology, 2024. available at https://csrc.nist.gov/Projects/pqc-dig-sig/round-2-additional-signatures.

[Zho25a] Yadi Zhong. ADC-BE: Optimizing Worst-Case Bandwidth in Broadcast Encryption with Boolean Functions. Cryptology ePrint Archive, Report 2025/650, 2025.

[Zho25b] Yadi Zhong. Variables for Free: Faulting MAYO via Valid Solutions. In *The International Conference on Artificial Intelligence Computing and Systems (AICompS)*, pages 1–8, 2025.

Appendix

Algorithm 1 $\mathsf{Sign}(M, esk)$

Input: Message M, expanded secret key esk.
Output: Signature $\sigma = (s, salt)$.

▷ *Decoder esk* ◁
$(seed_{sk}, O) \leftarrow esk$
$\{\boldsymbol{P}_i^{(1)}, \boldsymbol{L}_i\}_{i \in m} \leftarrow esk$
▷ *Hash message M, and derive salt and t* ◁
$M_digest \leftarrow \mathsf{SHAKE256}(M, digest_bytes)$
$R \leftarrow 0_{R_bytes}$ *or* $R \xleftarrow{\$} B^{R_bytes}$
$salt \leftarrow \mathsf{SHAKE256}(M_digest||R||seed_{sk}, salt_bytes)$
$t \leftarrow \mathsf{Decode}_{\mathsf{vec}}(m, \mathsf{SHAKE256}(M_digest||salt, \lceil m \log(q)/8 \rceil))$ ▷ $t \in \mathbb{F}_q^m$
▷ *Attempt to find a preimage for t.* ◁
for ctr from 0 to 255 **do**
  ▷ *Derive v_i and r.* ◁
  $V \leftarrow \mathsf{SHAKE256}(M_digest||salt||seed_{sk}||ctr)$
  $\{v_i\}_{i \in k} \leftarrow \mathsf{Decode}_{\mathsf{vec}}(n - o, V[i * v_bytes : (i+1) * v_bytes])$
  $r \leftarrow \mathsf{Decode}_{\mathsf{vec}}(ko, V[k * v_bytes : k * v_bytes + \lceil ko \log(q)/8 \rceil])$
  ▷ *Build linear system $Ax = y$.* ◁
  $A \leftarrow 0_{m \times ko} \in \mathbb{F}_q^{m \times ko}$
  $y \leftarrow t$, $\ell \leftarrow 0$
  $\{M_i\}_{i \in k} \leftarrow 0_{m \times o} \in \mathbb{F}_q^{m \times o}$
  $\left\{M_i[j, :]\right\}_{i \in k, j \in m} \leftarrow v_i^T \boldsymbol{L}_j$
  for i from 0 to $k - 1$ **do**
    for j from $k - 1$ to i **do**
      $$\mathbf{u} = \begin{cases} \{v_i^T P_a^{(1)} v_i\}_{a \in [m]} & \text{if } i = j \\ \{v_i^T P_a^{(1)} v_j + v_j^T P_a^{(1)} v_i\}_{a \in [m]} & \text{if } i \neq j \end{cases}$$
      $y \leftarrow y - E^\ell u$
      $A[:, i * o : (i+1) * o] \leftarrow A[:, i * o : (i+1) * o] + E^\ell M_j$
      if $i \neq j$ **then**
        $A[:, j * o : (j+1) * o] \leftarrow A[:, j * o : (j+1) * o] + E^\ell M_i$
      $\ell \leftarrow \ell + 1$
  ▷ *Try to solve the system.* ◁
  $x \leftarrow \mathsf{SampleSolution}(A, y, r)$
  if $x \neq \perp$ **then**
    break
▷ *Finish and output the signature.* ◁
$s \leftarrow 0_{kn}$
for i from 0 to $k - 1$ **do**
  $s[i * n : (i+1) * n] \leftarrow (v_i + \boldsymbol{O}x[i * o : (i+1) * o])||x[i * o : (i+1) * o]$
return $sig = \mathsf{Encode}_{\mathsf{vec}}(\mathsf{s})||salt$.

Fig. 1: MAYO Signature Sign Algorithm.

Algorithm 2 $\mathsf{SampleSolution}(A, y, r)$ Require $(ko \geq m)$

Input: Matrix $A \in \mathbb{F}^{m \times ko}$ $(ko \geq m)$, Target vector $y \in F_q^m$, Randomness $r \in F_q^{ko}$
Output: Solution $x \in F_q^n$ that satisfies $Ax = y$ if $\text{rank}(A) = m$; otherwise, output $\perp$.

▷ *Randomize the system using r.* ◁
$x \leftarrow r \in F_q^{ko}$
$y \leftarrow y - Ar$
▷ *Put (A y) in echelon form with leading 1's* ◁
$(A\ y) \leftarrow \mathsf{EF}(A\ y)$
▷ *Check if A has rank m* ◁
if $A[m-1, :] = 0_n$ **then**
  return $\perp$
▷ *Back-substitution* ◁
for r from $m-1$ to 0 **do**
  ▷ *Let c be the index of first non-zero element of $A[r, :]$.* ◁
  $x_c \leftarrow x_c + y_r$
  $y \leftarrow y - y_r A[:, c]$
return x

Fig. 2: MAYO Signature SampleSolution Algorithm.

Algorithm 3 $\mathsf{Verify}(epk, M, sig)$

Input: Expanded public key $epk \in B^{epk_byte}$, Message $M \in B^*$, Signature $sig \in B^{sig_bytes}$
Output: An integer result to indicate if sig is valid (result = 0) or invalid (result < 0).

▷ *Decode epk* ◁
$\{\boldsymbol{P}_i^{(1)}, \boldsymbol{P}_i^{(2)}, \boldsymbol{P}_i^{(3)}\}_{i \in m} \leftarrow epk$
▷ *Decode sig* ◁
$\{s, salt\} \leftarrow sig$
▷ *Hash message and derive t.* ◁
$M_digest \leftarrow \mathsf{SHAKE256}(M, digest_bytes)$
$t \leftarrow \mathsf{Decode}_{\mathsf{vec}}(m, \mathsf{SHAKE256}(M_digest||salt, \lceil mlog(q)/8 \rceil))$
▷ *Compute $P^*(s)$.* ◁
$y \leftarrow 0_m$
$\ell \leftarrow 0$
for i from 0 to $k-1$ **do**
  for j from $k-1$ to i **do**

$$\mathbf{u} = \begin{cases} \left\{ s_i^T \begin{pmatrix} P_a^{(1)} & P_a^{(2)} \\ \mathbf{0} & P_a^{(3)} \end{pmatrix} s_i \right\}_{a \in [m]} & \text{if } i = j \\ \left\{ s_i^T \begin{pmatrix} P_a^{(1)} & P_a^{(2)} \\ \mathbf{0} & P_a^{(3)} \end{pmatrix} s_j + s_j^T \begin{pmatrix} P_a^{(1)} & P_a^{(2)} \\ \mathbf{0} & P_a^{(3)} \end{pmatrix} s_i \right\}_{a \in [m]} & \text{if } i \neq j \end{cases}$$

    $y \leftarrow y + E^\ell u$
    $\ell \leftarrow \ell + 1$
▷ *Accept signature if $y = t$* ◁
if $y = t$ **then**
  return 0
return -1

Fig. 3: MAYO Signature Verification Verify Algorithm.