An improvement of algorithms to solve under-defined systems of multivariate quadratic equations [*]

Yasufumi Hashimoto [†]

Abstract

The problem of solving a system of multivariate quadratic equations over a finite field is known to be hard in general. However, there have been several algorithms of solving the system of quadratic equations efficiently when the number of variables is sufficiently larger than the number of equations (e.g., Kipnis et al., Eurocrypt 1999, Thomae-Wolf, PKC 2012, Cheng et al., PQCrypto 2014 and Furue et al., PQCrypto 2021). In the present paper, we propose a new algorithm which is available if the number of variables is smaller than that required in the previously given algorithms. We also analyze the security of MAYO, a variant of UOV, proposed in SAC 2021 and submitted to NIST's standardization project of additional digital signature schemes for Post-Quantum Cryptography.

Keywords. under-defined multivariate quadratic equations

1 Introduction

The *MQ problem*, the problem of solving a system of m multivariate quadratic polynomial equations of n variables over a finite field, is known to be a hard problem [11, 9] and is important for analyzing the security of multivariate public key cryptography (MPKC), which is a candidate for post-quantum cryptography.

The standard approaches to solving the MQ problem are by using the Gröbner basis (GB) algorithm, the XL algorithm and the hybrid approach, which is the combination of the exhaustive search and the GB/XL algorithms (see, e.g., [8, 5, 4]). These approaches have been widely used in the security analysis of MPKCs.

It is known that they are efficient especially for the over-defined systems ($m \gg n$). On the other hand, for under-defined systems ($n \gg m$), there have been several works to reduce the size of the systems by taking linear transforms of variables. In fact, Kipnis et al. [13] proposed a polynomial-time algorithm to reduce the problem of solving under-defined systems of quadratic equations with $n \geq m(m+1)$ over a finite field of even characteristic to the problem of solving a system of linear equations. Later, its polynomial-time algorithm has been arranged and improved by Courtois et al. [7], Miura et al. [14] and Cheng et al. [6].

While these algorithms will work in polynomial time, the required n is much larger than m and then it is not realistic to use them in the practical analysis of MPKCs. Thomae-Wolf [16] and Cheng et al. [6] proposed the algorithms to reduce the size of polynomial systems as

[*] This manuscript is a minor modification of [12].

[†] Department of Mathematical Science, University of the Ryukyus, hashimoto@math.u-ryukyu.ac.jp

arrangements of the algorithms given in [13] and [6], respectively. Although these algorithms do not solve in polynomial-time, the required n is not much larger than m. In this sense, they are more practical than the polynomial time algorithms given in [13, 7, 14, 6]. Recently, Furue et al. [10] improved the Thomae-Wolf algorithm slightly by considering the parameters of exhaustive search in the hybrid approach.

In the present paper, we propose a further improvement of the previously given algorithms [16, 6, 10] for solving under-defined systems, which is available for smaller n. We also give a security analysis of MAYO, a variant of UOV proposed in [2, 3] and submitted to NIST's standardization project of additional digital signature schemes for Post-Quantum Cryptography [15], and check that its security with given parameters is less than expected.

2 MQ problem

Let q be a power of prime and $\mathbf{F}_q$ a finite field of order q. For integers $n, m \geq 1$, denote by $F(\mathbf{x}) = {}^t(f_1(\mathbf{x}), \dots, f_m(\mathbf{x}))$ a set of m quadratic polynomials of n variables $\mathbf{x} = {}^t(x_1, \dots, x_n)$ over $\mathbf{F}_q$. We call the problem of solving the system $\{f_1(\mathbf{x}) = 0, \dots, f_m(\mathbf{x}) = 0\}$ of m quadratic equations of n variables the *MQ problem* and denote by $\mathrm{MQ}(q, n, m)$ the complexity of solving m quadratic equations of n variables over $\mathbf{F}_q$.

The Gröbner basis (GB) algorithm and the XL algorithm have been widely used to solve systems of polynomial equations and these are deeply related to each other [1]. Denote by $\mathrm{GB}(q, n, m)$ the complexity of the GB/XL algorithm for the system of m quadratic polynomials of n variables over $\mathbf{F}_q$. It is known that

$$\mathrm{GB}(q, n, m) \leq 3 \binom{n + d_{\mathrm{reg}}}{d_{\mathrm{reg}}}^2 \binom{n+2}{2} \tag{1}$$

if the system is semi-regular, where d_{reg} is the degree of the regularity of the corresponding polynomial system and is given as the smallest integer $d > 0$ such that the coefficient of t^d in the expansion of $\frac{(1-t^2)^m}{(1-t)^{n+1}}$ is not positive (see, e.g. [5]). This means that, if m is larger (n is smaller), the degree d_{reg} is smaller and then the complexity $\mathrm{GB}(q, n, m)$ is also smaller. It is why the GB/XL algorithm is efficient for over-defined ($m \gg n$) systems.

The *hybrid approach* [4] is the combination of the GB/XL algorithm and the exhaustive search to optimize the complexity. When $n = m$, it is described as follows. Let $k \geq 0$ be an integer and choose $u_1, \dots, u_k \in \mathbf{F}_q$ randomly. After that, solve the system of m equations $\{f_l(x_1, \dots, x_{m-k}, u_1, \dots, u_k) = 0\}_{1 \leq l \leq m}$ of $m - k$ variables by the GB/XL algorithm. If there exists a solution $(x_1, \dots, x_{m-k}) = (y_1, \dots, y_{m-k})$, $(y_1, \dots, y_{m-k}, u_1, \dots, u_k) \in \mathbf{F}_q^m$ is a solution of the original system. If there does not exist, choose another $(u_1, \dots, u_k) \in \mathbf{F}_q$ and try it again. Since the probability that it has a solution is considered to be about q^k, the complexity of the hybrid approach can be estimated by

$$\mathrm{MQ}(q, m, m) = \min_{k \geq 0} q^k \cdot \mathrm{GB}(q, m - k, m). \tag{2}$$

For under-defined systems ($n > m$), the parameter k is usually chosen to be $k \geq n - m$ and the complexity of the hybrid approach is similar to the system with $n = m$. However, if n is sufficiently larger than m, the system can be solved more efficiently. For example, the following four works proposed polynomial-time algorithms of solving under-defined systems.

Kipnis et al. [13]	q: even	$n \geq m(m+1)$
Courtois et al. [7]	q: any	$n \geq 2^{m/7} m(m+1)$
Miura et al. [14]	q: even	$n \geq \frac{1}{2}m(m+3)$
Cheng et al. [6]	q: any	$n \geq \frac{1}{2}m(m+1)$

These algorithms require much larger n than m and are not considered to be effective for analyzing practical multivariate signature schemes. The following two works proposed algorithms available for smaller n and with the complexity $\mathrm{MQ}(q, m-a, m-a)$, which is not in polynomial-time but is smaller than $\mathrm{MQ}(q, m, m)$.

Thomae-Wolf [16]	q: even	$n \geq (a+1)m$
Cheng et al. [6]	q: any	$n \geq (a+1)(m - \frac{1}{2}a)$

Recently, Furue et al. [10] improved Thomae-Wolf's algorithm by considering the parameter k of exhaustive search in the hybrid approach. It is available if q is even and

$$n \geq (a+1)(m-k) + k,$$

and its complexity is

$$q^k \cdot \mathrm{MQ}(q, m-a-k, m-a). \tag{3}$$

In the present paper, we propose a further improvement of the algorithms given above. Our algorithm is available for any q and

$$n \geq \max\left\{(a+1)(m-k-a+1), a(m-k) - (a-1)^2 + k\right\} \tag{4}$$

and its complexity is

$$q^k \cdot \Big(\mathrm{MQ}(q, m-a-k, m-a) + \mathrm{MQ}(q, a-1, a-1)\Big) + (m-a-k+1) \cdot \mathrm{MQ}(q, a, a). \tag{5}$$

Note that, if a is small, its complexity is almost the same to (3) with the same a, k. In the following table, we compare the required n for small a.

Table 1: Comparison of required n

a	TW [16]	C. [6]	F. [10]	This work
1	$2m$	$2m-1$	$2m-k$	$2m-2k$
2	$3m$	$3m-3$	$3m-2k$	$3m-3k-3$
3	$4m$	$4m-6$	$4m-3k$	$4m-4k-8$
4	$5m$	$5m-10$	$5m-4k$	$5m-5k-15$
5	$6m$	$6m-15$	$6m-5k$	$6m-6k-24$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

3 The approach of Furue et al.

We now describe the approach of Furue et al. [10] for even q, which consists of the following three steps.

Step 1. Find an $(n-m+k)\times(m-k)$ matrix M such that, for $1\le l\le a$,

$$\begin{aligned}\bar{f}_l(\mathbf{x}) :=& f_l\left(\begin{pmatrix} I_{m-k} & \\ M & I_{n-m+k}\end{pmatrix}\mathbf{x}\right)\\ =& K_l(x_1^2,\dots,x_{m-k}^2)+\sum_{i=1}^{m-k} x_i\cdot L_{li}(x_{m-k+1},\dots,x_n)+Q_l(x_{m-k+1},\dots,x_n)\\ =& {}^t\mathbf{x}\left(\begin{array}{ccc|c} * & & & \\ & \ddots & & * \\ & & * & \\ \hline & * & & *_{n-m+k}\end{array}\right)\mathbf{x}+(\text{linear polynomial of }\mathbf{x}),\end{aligned}$$

where K_l, L_{li} are linear polynomials and Q_l is a quadratic polynomial.

Step 2. Choose $u_1,\dots,u_{n-m+k}\in\mathbf{F}_q$ such that

$$L_{li}(u_1,\dots,u_{n-m+k})=0$$

for $1\le l\le a$ and $1\le i\le m-k$.

Step 3. Solve the system

$$\{\bar{f}_l(x_1,\dots,x_{m-k},u_1,\dots,u_{n-m+k})=0\}_{1\le l\le m} \tag{6}$$

of m equations of $m-k$ variables $(x_1,\dots,x_{m-k})$. If there exists a solution of (6), output

$$\begin{pmatrix} I_{m-k} & \\ -M & I_{n-m+k}\end{pmatrix}{}^t(x_1,\dots,x_{m-k},u_1,\dots,u_{n-m+k})$$

as a solution of $\{f_l(\mathbf{x})=0\}_{1\le l\le m}$. If not, go back to Step 2 and choose another $(u_1,\dots,u_{n-m+k})$.

In Step 1, to find a matrix M, fix the first column of M to be zero and fix the second column of M such that the coefficients of x_1x_2 in the quadratic polynomials $\bar{f}_1(\mathbf{x}),\dots,\bar{f}_a(\mathbf{x})$ are zero. Next fix the third column of M such that the coefficients of x_1x_3, x_2x_3 are zero. Similarly, fix the columns of M in turn and finally fix the $(m-k)$-th column such that the coefficients of $x_1x_{m-k},\dots,x_{m-k-1}x_{m-k}$ are zero. Note that, to fix the i-th column, we need to solve a system of $a(i-1)$ equations of $n-m+k$ variables and then the condition $n\ge(a+1)(m-k)-a$ is required in Step 1.

In Step 2, we need to solve $a(m-k)$ equations of $n-m+k$ variables. Remark that, since the probability that the system (6) in Step 3 has a solution is considered to be about q^{-k}, we need additional k parameters in this step. Then the condition $n\ge(a+1)(m-k)+k$ is required.

In Step 3, the first a equations in (6) are linear equations

$$K_l(x_1^2,\dots,x_{m-k}^2)=(\text{const.}), \tag{7}$$

of $x_1^2,\dots,x_{m-k}^2$, and the remaining $m-a$ equations are quadratic equations of $x_1,\dots,x_{m-k}$. Since, by taking $q/2$ power, we can transform the equations in (7) into linear equations of

$x_1, \ldots, x_{m-k}$ [16, 10]. This means that the problem of solving the system (6) can be reduced to the problem of solving the system of $m-a$ equations of $m-a-k$ variables. Since the probability that such a system has a solution is considered to be about q^{-k}, we need to repeat Step 2 and 3 at most about q^k times. We thus conclude that the condition $n \geq (a+1)(m-k)+k$ is required in this approach and the complexity is $q^k \cdot \mathrm{MQ}(q, a-k, a)$.

4 Proposed algorithm

In this section, we describe our new algorithm.

Step 1. Find an $(n-m+a+k-1) \times (m-a-k+1)$ matrix M_1 such that, for $1 \leq l \leq a$,

$$\begin{aligned}
\tilde{f}_l(\mathbf{x}) :=& f_l \left(\begin{pmatrix} I_{m-a-k+1} & \\ M_1 & I_{n-m+a+k-1} \end{pmatrix} \mathbf{x} \right) \\
=& \sum_{i=1}^{m-a-k+1} x_i \cdot \tilde{L}_{li}(x_{m-a-k+2}, \ldots, x_n) + \tilde{Q}_l(x_{m-a-k+2}, \ldots, x_n) \\
=& {}^t\mathbf{x} \left(\begin{array}{cc|c} 0_{m-a-k+1} & * & * \\ * & *_{a-1} & * \\ \hline * & * & *_{n-m+k} \end{array} \right) \mathbf{x} + (\text{linear polynomial of } \mathbf{x}),
\end{aligned}$$

where $\tilde{L}_{li}$ is a linear polynomial and $\tilde{Q}_l$ is a quadratic polynomial.

Step 2. Find an $(n-m+k) \times (a-1)$ matrix M_2 such that, for $1 \leq l \leq a-1$,

$$\begin{aligned}
\bar{f}_l(\mathbf{x}) :=& \tilde{f}_l \left(\begin{pmatrix} I_{m-a-k+1} & & \\ & I_{a-1} & \\ & M_2 & I_{n-m+k} \end{pmatrix} \mathbf{x} \right) \\
=& P_l(x_{m-a-k+2}, \ldots, x_{m-k}) + \sum_{i=1}^{m-k} x_i \cdot L_{li}(x_{m-k+1}, \ldots, x_n) + Q_l(x_{m-k+1}, \ldots, x_n) \\
=& {}^t\mathbf{x} \left(\begin{array}{cc|c} 0_{m-a-k+1} & 0 & * \\ 0 & *_{a-1} & * \\ \hline * & * & *_{n-m+k} \end{array} \right) \mathbf{x} + (\text{linear polynomial of } \mathbf{x}),
\end{aligned}$$

where L_{li} is a linear polynomial and P_l, Q_l are quadratic polynomials.

Step 3. Choose $u_1, \ldots, u_{n-m+k} \in \mathbf{F}_q$ such that

$$L_{li}(u_1, \ldots, u_{n-m+k}) = 0$$

for $1 \leq l \leq a-1$ and $1 \leq i \leq m-a-k+1$.

Step 4. Solve the system

$$\left\{ \bar{f}_l(x_1, \ldots, x_{m-k}, u_1, \ldots, u_{n-m+k}) = 0 \right\}_{1 \leq l \leq a-1} \tag{8}$$

of $a-1$ quadratic equations of $a-1$ variables $(x_{m-a-k+2}, \ldots, x_{m-k})$. Let $(v_1, \ldots, v_{a-1})$ be its solution.

Step 5. Solve the system

$$\left\{\bar{f}_l(x_1,\dots,x_{m-a-k+1},v_1,\dots,v_{a-1},u_1,\dots,u_{n-m+k})=0\right\}_{a\leq l\leq m} \tag{9}$$

of $m-a+1$ equations of $m-a-k+1$ variables $(x_1,\dots,x_{m-a-k+1})$. If there exists a solution of (9), output $\left(\begin{smallmatrix} I_{m-a-k+1} & & \\ M_{11} & I_{a-1} & \\ M_{12} & M_2 & I_{n-m+k}\end{smallmatrix}\right)^{-1} {}^t(x_1,\dots,x_{m-a-k+1},\, v_1,\dots,v_{a-1},u_1,\dots,u_{n-m+k})$ as a solution of the original system $\{f_l(\mathbf{x})=0\}_{1\leq l\leq m}$, where M_{11},M_{12} are given by $M_1=\left(\begin{smallmatrix}M_{11}\\ M_{12}\end{smallmatrix}\right)$. If not, go back to Step 3 and choose another $(u_1,\dots,u_{n-m+k})$.

In Step 1, fix the first column of M_1 such that the coefficients of x_1^2 in $\tilde{f}_1(\mathbf{x}),\dots,\tilde{f}_a(\mathbf{x})$ are zero. Next fix the second column of M_1 such that the coefficients of x_1x_2,x_2^2 are zero. Similarly, fix the columns of M in turn and finally fix the $(m-a-k+1)$-th column such that the coefficients of $x_1x_{m-a-k+1},\dots,x_{m-a-k+1}^2$ are zero. Note that, to fix the i-th column of M_1, we need to solve a system of a quadratic equations and $a(i-1)$ linear equations of $n-m+a+k-1$ variables. This means that this step requires the condition $n\geq(a+1)(m-a-k+1)$ and the complexity $(m-a-k+1)\cdot\mathrm{MQ}(q,a,a)$.

In Step 2, fix the first column of M_2 such that the coefficients of $x_1x_{m-a-k+2},\dots,$ $x_{m-a-k+1}x_{m-a-k+2}$ in $\bar{f}_1(\mathbf{x}),\dots,\bar{f}_{a-1}(\mathbf{x})$ are zero. Next fix the second column of M_2 such that the coefficients of $x_1x_{m-a-k+3},\dots,x_{m-a-k+1}x_{m-a-k+3}$ are zero. Similarly, fix the columns of M_2 in turn and finally fix the $(a-1)$-th column such that the coefficients of $x_1x_{m-k},\dots,x_{m-a-k+1}x_{m-k}$ are zero. Note that, to fix the i-th column of M_2, we need to solve a system of $(a-1)(m-a-k+1)$ linear equations of $n-m+k$ variables. This means that this step requires the condition $n\geq a(m-k)-(a-1)^2$.

In Step 3, we need to solve $(a-1)(m-a-k+1)$ linear equations of $n-m+k$ variables. Remark that, since the probability that the system (9) in Step 5 has a solution is considered to be about q^{-k}, we need additional k parameters in this step. Then the condition $n\geq a(m-k)-(a-1)^2+k$ is required.

In Step 4, we solve the system (8) of $a-1$ quadratic equations of $a-1$ variables. Then the complexity of this step is $\mathrm{MQ}(q,a-1,a-1)$.

In Step 5, we solve the system (9) of $m-a+1$ equations and $m-a-k+1$ variables. Since the equation $\bar{f}_a=0$ in the system (9) is linear and the others are quadratic, the complexity of solving (9) is $\mathrm{MQ}(q,m-a-k,m-a)$. Since the probability that such a system has a solution is considered to be about q^{-k}, we need to repeat Step 3-5 at most about q^k times. We thus conclude that our new approach requires the condition

$$n\geq\max\{(a+1)(m-k-a+1),a(m-k)-(a-1)^2+k\}$$

and the complexity is

$$(m-a-k+1)\cdot\mathrm{MQ}(q,a,a)+q^k\cdot\left(\mathrm{MQ}(q,a-1,a-1)+\mathrm{MQ}(q,m-a-k,m-a)\right).$$

5 Security of MAYO

MAYO proposed by Beullen in 2021 [2] is a variant of UOV with small keys. In the parameters selected in [2], the numbers of variables are more than ten times of the numbers of polynomials, and then the security against the algorithms for under-defined systems should be studied quite carefully. Table 2 describes the selected parameters (q, n, m) in [2] and the security against the direct attack with the Thomae-Wolf approach, the Kipnis-Shamir attack, the reconciliation attack and the intersection attack. In this section, we study and compare the security of MAYO in the table above against the approaches of Thomae-Wolf [16], Cheng et al. [6], Furue et al. [10] and the proposed approach described in §4 under the assumption that the systems of equations to be solved are semi-regular and that the complexity of the GB/XL algorithm is as given in (1).

Table 2: Selected parameters of MAYO in [2]

No.	(q, n, m)	Security (bits)
1	$(16, 924, 67)$	143
2	$(16, 938, 68)$	146
3	$(16, 1666, 99)$	207
4	$(16, 1980, 102)$	207
5	$(16, 2470, 132)$	273
6	$(16, 2489, 132)$	273

We first study the first parameter $(q, n, m) = (16, 924, 67)$. We see that $a = 12$ satisfies the condition of the approach of Thomae-Wolf [16] and then one can solve such a system of quadratic equations with the complexity $\mathrm{MQ}(16, 55, 55) \risingdotseq 2^{143}$. For the approach of Cheng et al. [6], $a = 14$ satisfies the condition and then one can solve with the complexity $\mathrm{MQ}(16, 53, 53) \risingdotseq 2^{138}$. Furthermore, $(a, k) = (16, 15)$ satisfies the condition of the approach of Furue et al. [10] and then the complexity of its approach is $16^{15} \cdot \mathrm{MQ}(16, 36, 51) \risingdotseq 2^{135}$.

In our approach, $(a, k) = (26, 8)$ satisfies the condition and then the complexity is $34 \cdot \mathrm{MQ}(16, 26, 26) + 16^8 \cdot (\mathrm{MQ}(16, 25, 25) + \mathrm{MQ}(16, 33, 41)) \risingdotseq 2^{110}$. Table 3 summarizes the security (bits) against the corresponding approaches. It shows that our new approach will reduce the security of MAYO against the direct attack by more than 30 bits.

Table 3: Security (bits) of MAYO in [2]

No.	T-W [16]	C. [6]	F. [10]	This work
1	143	138	135	110
2	146	140	137	113
3	207	201	197	170
4	207	201	197	162
5	273	265	262	237
6	273	267	262	234

Recently, new parameter sets of MAYO [3] was proposed as a first round candidate of NIST's standardization project of additional digital signature schemes for Post-Quantum Cryptography [15]. Table 4 summarizes the parameter sets and the security of MAYO given in [3]. In this table, "Security" is the minimum of their security bits described in Table 5.1 of [3] against Kipnis-Shamir's attack, the reconciliation attack, the intersection attack, the direct attack with the approach of Furue et al. [10] and the claw finding attack. This shows that our approach will reduce the security of MAYO_1 (and MAYO_3, MAYO_5 slightly).

Table 4: New parameter sets of MAYO in [3]

	(q, n, m)	Security	This work
MAYO$_1$	$(16, 594, 64)$	143	126
MAYO$_2$	$(16, 304, 64)$	143	151
MAYO$_3$	$(16, 1000, 96)$	207	200
MAYO$_5$	$(16, 1596, 128)$	272	263

6 Conclusion

In this paper, we propose a new algorithm to solve an under-defined system of multivariate quadratic equations over a finite field. The number of variables required in our approach is less than those in the previous works. This means that the condition for solving under-defined systems is relaxed, and then the parameters should be chosen more carefully than before when building a multivariate signature scheme with under-defined systems.

Acknowledgment. The author would like to thank the anonymous reviewer(s) for the journal version [12] of this paper. He was supported by JSPS Grant-in-Aid for Scientific Research (C) no.17K05181, no.22K03234 and JST Crest no.JPMJCR2113.

References

[1] G. Ars, J.-C Faugére, H. Imai, M. Kawazoe, M. Sugita, Comparison between XL and Gröbner basis algorithms, Asiacrypt'04, LNCS **3329**, pp.338–358.

[2] W. Beullens, MAYO: Practical post-quantum signatures from Oil-and-Vinegar maps, SAC'21, LNCS **13203**, pp.355–376, extended in `https://eprint.iacr.org/2021/1144` (version at Dec. 23, 2022).

[3] W. Beullens, F. Campos, S. Celi, B. Hess, M. J. Kannwischer, MAYO, `https://pqmayo.org/assets/specs/mayo.pdf` (version at June 1, 2023).

[4] L. Bettale, J.C. Faugère, L. Perret, Solving polynomial systems over finite fields: Improved analysis of the hybrid approach, ISSAC'12, pp.67–74.

[5] C.M. Cheng, T. Chou, R. Niederhagen, B.Y. Yang Solving Quadratic Equations with XL on Parallel Architectures, CHES'12, LNCS **7428**, pp.356–373.

[6] C.M. Cheng, Y. Hashimoto, H. Miura, T. Takagi, A polynomial-time algorithm for solving a class of underdetermined multivariate quadratic equations over fields of odd characteristics, PQCrypto'14, LNCS **8772**, pp.40–58.

[7] N. Courtois, L. Goubin, W. Meier, J.-D. Tacier, Solving underdefined systems of multivariate quadratic equations, PKC'02, LNCS **2274**, pp.211–227.

[8] J.C. Faugére, A new efficient algorithm for computing Gröbner bases (F4), J. Pure Appl. Algebra **139** (1999), pp.61–88.

[9] A.S. Fraenkel, Y. Yesha, Complexity of problems in games, graphs and algebraic equations. Discrete Appl. Math. **1** (1979), pp.15–30.

[10] H. Furue, S. Nakamura, T. Takagi, Improving Thomae-Wolf algorithm for solving underdetermined multivariate quadratic polynomial problem, PQCrypto'21, LNCS **12841**, pp.65–78.

[11] M.R. Garey, D.S. Johnson, Computers and Intractability, A Guide to the Theory of NP-completeness, W.H. Freeman, San Francisco, 1979.

[12] Y. Hashimoto, An improvement of algorithms to solve under-defined systems of multivariate quadratic equations, LSIAM Letters, **15** (2023), to appear.

[13] A. Kipnis, J. Patarin, L. Goubin, Unbalanced oil and vinegar signature schemes, Eurocrypt'99, LNCS **1592**, pp.206–222, extended in `http://www.goubin.fr/papers/OILLONG.PDF`, 2003.

[14] H. Miura, Y. Hashimoto, T. Takagi, Extended algorithm for solving underdefined multivariate quadratic equations, PQCryoto'13, LNCS **7932**, pp.118–135.

[15] NIST, Post-Quantum Cryptography: Digital Signature Schemes, Round 1 Additional Signatures, `https://csrc.nist.gov/Projects/pqc-dig-sig/round-1-additional-signatures`.

[16] E. Thomae, C. Wolf, Solving underdetermined systems of multivariate quadratic equations revisited, PKC'12, LNCS **7293**, pp.156–171.