A Hybrid Method for Solving the Minrank Problem

Yacheng Wang[1,a)] Yasuhiko Ikematsu[2,b)] Shuhei Nakamura[3,c)]
Tsuyoshi Takagi[1,d)]

Abstract: The minrank problem is often considered in the cryptanalysis of multivariate public key cryptography (MPKC) and code-based cryptography. There have been many multivariate cryptosystems proven insecure due to their weakness against the minrank attack, which is an attack that transforms breaking a cryptosystem into solving a minrank problem instance. In this paper, we review two methods, the Kipnis-Shamir method, and minors modeling for solving a minrank instance, and then propose a hybrid method that combines these two modeling methods. Our new method manages to avoid the disadvantages of the Kipnis-Shamir method and minors modeling, and it at least is as effective as the Kipnis-Shamir method. Moreover, we consider the proposed hybrid method with different types of variables specified, from which possible improvements can be brought to the proposed hybrid method. We also apply our hybrid method on one of NIST Post-quantum cryptography round 2 submissions, Rainbow.

Keywords: Minrank Problem, Multivariate Cryptography, Gröbner basis.

1. Introduction

With currently widely used cryptosystems, RSA [25] and ECC [20], being threatened by the development of quantum computers because of Shor's factoring algorithm [26], research on the post-quantum cryptography has become more urgent. NIST [1], [10] anticipated a realization of quantum computers that are capable enough of breaking 2048-bit RSA by the year of 2030, and they have taken actions on standardizing post-quantum cryptosystems.

Among all candidates of post-quantum cryptosystems, multivariate public key cryptosystems and code-based cryptosystems often face some challenges from a so-called minrank attack, that is an attack that transforms breaking a cryptosystem into solving a minrank problem instance. The minrank problem asks one to find a linear combination of given m matrices $M_1, \ldots, M_m$, that has rank between 1 and r. This problem is proven to be an NP-complete problem [7], and by far, there are three different methods proposed for solving it, that are the Kipnis-Shamir method [19], minors modeling [4] and linear algebra search method [18].

In multivariate public-key cryptography, many attempts on building secure cryptosystems failed due to their weakness against the minrank attack, for example, HFE [19], SRP [23], ZHFE [8], and TTM [18]. Techniques such as enlarging parameters or applying modifiers are also applied to some multivariate cryptosystems such as Rainbow [12] and HFEv- [22], [24] because of the minrank attack.

Unlike fairly well-understood minors modeling and linear algebra search method, there were not many results published on the complexity analysis of the Kipnis-Shamir method until Verbel et al. [27] gave their analysis. They gave a method of constructing non-trivial syzygies, and hence understanding the first fall degree of the polynomial system obtained from the Kipnis-Shamir method, which indicates a tighter complexity bound. The usage of overdetermined subsystems was also pointed out for improvements. As its advantages, the Kipnis-Shamir method is simple and gives low first fall degrees. As its drawback, it introduces many new variables. On the other hand, as for minors modeling, a thorough analysis of its complexity is given [9], [13], [17]. This method does not introduce new variables but requires the computation of many minors.

To avoid drawbacks lie in the Kipnis-Shamir method and minors modeling, we propose a new hybrid method that merges these two methods. Our new method solves the minrank problem by using a subsystem from Kipnis-Shamir method and a subsystem from minors modeling. When determined or overdetermined subsystems from the Kipnis-Shamir method are used in our hybrid method, the complexity is the same as only using determined or overdetermined subsystem from the Kipnis-Shamir method. Therefore, our

[1] Department of Mathematical Informatics, University of Tokyo
[2] Institute of Mathematics for Industry, Kyushu University
[3] Department of Liberal Arts and Basic Sciences, Nihon University
a) yacheng_wang@mist.i.u-tokyo.ac.jp
b) ikematsu@imi.kyushu-u.ac.jp
c) nakamura.shuhei@nihon-u.ac.jp
d) takagi@mist.i.u-tokyo.ac.jp

method can either improve the Kipnis-Shamir method or have the same complexity as the Kipnis-Shamir method without computing as many minors as minors modeling. Another contribution of this paper is to consider the difference of significance in two sets of variables in a bilinear system (see §2.2) when we adopt a hybrid approach [3] to solving multivariate polynomials. The hybrid approach is a combination of exhaustive search and Gröbner basis computing for solving a set of multivariate polynomials. The more significant set of variables being specified expects to bring more degree drops in the first fall degree and the solving degree of a polynomial system.

The paper is organized as follows. Section 2 introduces some basic knowledge about solving a set of multivariate polynomials and bilinear systems. In section 3, we review the minrank problem as well as the Kipnis-Shamir method and minors modeling, which are methods for solving the minrank problem. In section 4, we propose a hybrid method for solving the minrank problem and discuss the behavior of the hybrid method with some variables specified. In section 5, we present experimental results and application of our method on a multivariate signature scheme, Rainbow. Finally, we give a conclusion in Section 6.

2. Multivariate Quadratic Problem and Bilinear Systems

2.1 Multivariate Quadratic Problem

Let $\mathbb{F}$ be a finite field of order q, $m, n \in \mathbb{N}$ be positive integers and $R := \mathbb{F}[x_1, \ldots, x_n]$ be the polynomial ring in variables $x_1, \ldots, x_n$ over $\mathbb{F}$. Then the multivariate quadratic (MQ) problem is defined as follows.

Problem 1 (Multivariate Qudratic (MQ) Problem)**.** *Given a set of multivariate quadratic polynomials $f_1, \ldots, f_m \in R$ and $\mathbf{y} = (y_1, \ldots, y_m) \in \mathbb{F}^m$, find $(z_1, \ldots, z_n) \in \mathbb{F}^n$ such that $f_1(z_1, \ldots, z_n) = y_1, \ldots, f_m(z_1, \ldots, z_n) = y_m$.*

The currently most effective method for solving this problem is through Gröbner basis [6] computing. Efficient algorithms for computing a Gröbner basis include XL [11], F4 [14], and F5 [15]. A good indicator of the complexity of computing a Gröbner basis is the degree of regularity, which is defined as follows.

Definition 1 (Degree of regularity (d_{reg}) [2])**.** *Let $f_1, \ldots, f_m \in R$ be polynomials. Denote the homogeneous component of the highest degree of $f_1, \ldots, f_m$ by $f'_1, \ldots, f'_m$, and let $\mathcal{I}_R = \langle f_1, \ldots, f_m \rangle$, $\mathcal{I}'_R = \langle f'_1, \ldots, f'_m \rangle$ be ideals of R. Then the* degree of regularity *of $\mathcal{I}_R$ is defined by*

$$d_{reg}(\mathcal{I}_R) = \min \left\{ d \geq 0 \mid \dim_{\mathbb{F}} \langle f \in \mathcal{I}'_R, \deg(f) = d \rangle = \binom{n+d-1}{d} \right\}.$$

Besides this degree of regularity, we also have other two indicators that are often used in estimating the complexity of a Gröbner basis computing algorithm, which are called the solving degree (say d_{sol}) and the first fall degree (say d_{ff}).

The solving degree is the maximum polynomial degree that appears in the process of computing a Gröbner basis for the ideal $\mathcal{I}_R$. To define the first fall degree, we need to be familiar with a notion called non-trivial syzygies. Denote the polynomial ring $\mathbb{F}[x_1, \ldots, x_n]$ by R, and define $R_{\leq d}$ to be the set of polynomials of degree d or less. Let $f_1, \ldots, f_m$ be polynomials in R of degree $d_1, \ldots, d_m$, respectively, we define the following homomorphism:

$$\begin{aligned} \sigma_d(f_1, \ldots, f_k) : \ \oplus_{i=1}^k R_{\leq d-d_i} &\to R_{\leq d} \\ (m_1, \ldots, m_k) &\mapsto \sum_{i=1}^k m_i f_i. \end{aligned}$$

Then the elements $(m_1, \ldots, m_k) \in \oplus_{i=1}^k R_{\leq d-d_i}$ that satisfy $\sum_{i=1}^k m_i f_i = 0$ are called syzygies of degree d. For exmaple, $(f_2, -f_1, 0, \ldots, 0)$ is a syzygy of degree d. More specifically, the combinations of k-tuples in $\oplus_{i=1}^k R_{\leq d-d_i}$ with $m_i = f_j, m_j = -f_i$ for some i, j and $m_t = 0$ for $t \neq i, j$ are called *trivial syzygies*. The syzygies that are not in the linear span of the trivial syzygies are called *non-trivial syzygies*. Non-trivial syzygies account for the non-trivial degree falls during a Gröbner basis computation.

Definition 2 (First fall degree (d_{ff}))**.** *Let $f_1, \ldots, f_m \in R$ be polynomials, their homogeneous component of the highest degree be $f'_1, \ldots, f'_m$. Then the* first fall degree *of the polynomials $f_1, \ldots, f_m$ is defined as the smallest degree d at which a degree d non-trivial syzygy of the polynomials $f'_1, \ldots, f'_m$ exists.*

It is widely accepted that d_{ff} can serve as a good indicator for the complexity of computing a Gröbner basis since experimentally it is usually very close to d_{reg}.

2.2 Bilinear System

A bilinear polynomial is defined as follows.

Definition 3 (Bilinear polynomial)**.** *Let*

$$\mathbf{x} = (x_1, \ldots, x_{n_1}), \ \mathbf{y} = (y_1, \ldots, y_{n_2})$$

be variables, $\mathbb{F}[\mathbf{x}, \mathbf{y}]$ be the polynomial ring in $\mathbf{x}$ and $\mathbf{y}$ over a field $\mathbb{F}$. A bilinear polynomial *$f \in \mathbb{F}[\mathbf{x}, \mathbf{y}]$ is a quadratic polynomial, and affine in each set of variables, i.e. $\deg_{\mathbf{x}}(f) = \deg_{\mathbf{y}}(f) = 1$.*

Regarding a set of bilinear polynomials, there are some special properties, and we will use the Jacobian matrix to explain these properties. The Jacobian matrix of a set of polynomials is defined as follows.

Definition 4 (Jacobian matrix)**.** *Given a sequence of bilinear polynomials $F = (f_1, \ldots, f_m) \in \mathbb{F}[\mathbf{x}, \mathbf{y}]^m$, then the* Jacobian matrices *of F with respect to variables $\mathbf{x}$ and $\mathbf{y}$ are given by*

$$jac_{\mathbf{x}}(F) = \left[\frac{\partial f_i}{\partial x_j} \right], \quad jac_{\mathbf{y}}(F) = \left[\frac{\partial f_i}{\partial y_k} \right],$$
$$(1 \leq i \leq m, 1 \leq j \leq n_1, 1 \leq k \leq n_2).$$

And we have the following proposition for a set of homogeneous bilinear polynomials:

Proposition 1. *Let $F = (f_1, \ldots, f_m) \in \mathbb{F}[\mathbf{x}, \mathbf{y}]^m$ be a sequence of homogeneous bilinear polynomials, then the following statements hold :*

(i) If $G = (g_1, \ldots, g_m) \in \mathbb{F}[\mathbf{y}]^m$, then G is a non-trivial syzygy of F if and only if $G \cdot jac_{\mathbf{x}}(F) = 0$.

(ii) Similar statement holds for $jac_{\mathbf{y}}(F)$.

Proof. ($\Rightarrow$) Since we have $jac_{\mathbf{x}}(F)\mathbf{x} = F^\top$, and G is a non-trivial syzygy, we obtain $\sum_{i=1}^{m} g_i f_i = 0$. Replacing f_i by $F^\top$ gives us $G \cdot jac_{\mathbf{x}}(F)\mathbf{x} = 0$.

($\Leftarrow$) Conversely, given $G \cdot jac_{\mathbf{x}}(F) = 0$, we easily obtain $\sum_{i=1}^{m} g_i f_i = 0$, which implies G is a syzygy. Since G is in $\mathbb{F}[\mathbf{y}]$, not $\mathbb{F}[\mathbf{x}, \mathbf{y}]$, we know G is a non-trivial syzygy of F.

Similar proof can be applied to $jac_{\mathbf{y}}(F)$ case. □

From the above proposition, we can construct some non-trivial syzygies of a set of homogeneous bilinear polynomials with the left kernel of its Jacobian matrices with respect to each set of variables. These Jacobian matrices have linear polynomials as its entries, and we need to compute their left kernels. To inspire this computation, we give two examples.

Example 1. *Consider solving* $\begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0.$

We convert it to the echelon form : $\begin{bmatrix} a_1 & a_2 & a_3 \\ 0 & \frac{b_2a_1 - b_1a_2}{a_1} & \frac{b_3a_1 - b_1a_3}{a_1} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0$. *Let* $x_3 = t$, *then* $x_2 = -t\left(\frac{b_3a_1 - b_1a_3}{b_2a_1 - b_1a_2}\right)$, $x_1 = t\left(\frac{b_2a_3 - b_3a_2}{b_2a_1 - b_1a_2}\right)$. *If we reparametrize* $x_3 = t\begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix}$ *we finally obtain* $\frac{x_1}{\begin{vmatrix} a_3 & a_2 \\ b_3 & b_2 \end{vmatrix}} = \frac{-x_2}{\begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix}} = \frac{x_3}{\begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix}} = t.$

Example 2. *Consider solving* $\begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0.$

We convert it to the echelon form : $\begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ 0 & \frac{b_2a_1 - b_1a_2}{a_1} & \frac{b_3a_1 - b_1a_3}{a_1} & \frac{b_4a_1 - b_1a_4}{a_1} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0$. *Let* $x_3 = t, x_4 = s$. *Then we have* $x_1 = -\begin{vmatrix} a_2 & a_4 \\ b_2 & b_4 \end{vmatrix} s - \begin{vmatrix} a_2 & a_3 \\ b_2 & b_3 \end{vmatrix} t$, $x_2 = \begin{vmatrix} a_1 & a_4 \\ b_1 & b_4 \end{vmatrix} s + \begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix} t$, $x_3 = -\begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} t$, $x_4 = -\begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} s$.

From these two examples, we know that the kernel of a matrix with linear polynomial entries have elements in the span of its maximal minors. Here, maximal minor refers to determinants of square submatrices of the maximal size of a matrix.

3. Minrank Problem

In this section, we introduce the minrank problem as well as two existing methods for solving the minrank problem, the Kipnis-Shamir method, and minors modeling.

3.1 Minrank Problem

The minrank problem is given as follows.

Problem 2 (Minrank Problem)**.** *Given a field $\mathbb{F}$ of q elements, a positive integer $r \in \mathbb{N}$ and $n \times n$ matrices $M_1, \ldots, M_m \in \mathbb{F}^{n \times n}$, find $x_1, \ldots, x_m \in \mathbb{F}$ such that*

$$0 < Rank\left(\sum_{i=1}^{m} x_i M_i\right) \leq r.$$

Depending on the relation of the three parameters, the minrank problem can be classified into under-defined, well-defined and over-determined cases.

Definition 5. *If $m > (n-r)^2$, a minrank instance is called under-defined; if $m = (n-r)^2$, a minrank instance is called well-defined; otherwise, it is called over-determined.*

3.2 Minors Modeling

Minors modeling is based on the fact that all $(r+1) \times (r+1)$ minors of $\left(\sum_{i=1}^{m} x_i M_i\right)$ vanish at $(x_1, \ldots, x_m)$ when $\left(\sum_{i=1}^{m} x_i M_i\right)$ has rank less than r. This method gives us a system of $\binom{n}{r+1}^2$ equations in m variables. The property of this polynomial system is related to the so-called determinantal ideal. In [9], [13], [17], intensive analysis on the property of the ideal generated by polynomials from minors modeling is given. Minors modeling, as its advantage, does not introduce any extra variables and gives an over-determined polynomial system. But it also means we have to compute as many as $\binom{n}{r+1}^2$ minors of a matrix with linear polynomial entries, and sometimes they are not entirely necessary. For example, when $r \ll n$, we will have a huge amount of equations, and if we use partial equations, the solving process may take less time than using full equations.

3.3 The Kipnis-Shamir Method

The Kipnis-Shamir method [19] was first used to break the HFE cryptosystem [21]. This method is based on the fact that the dimension of the kernel of $\left(\sum_{i=1}^{m} x_i M_i\right)$ should be larger than or equal to $n - r$ since it has rank smaller than or equal to r. If we assume a basis matrix of this kernel subspace is $\begin{bmatrix} I_{n-r} \\ K \end{bmatrix}$, where I_{n-r} is an identity matrix of size $n - r$ and K is an $r \times (n-r)$ matrix, then we have

$$\left(\sum_{i=1}^{m} x_i M_i\right) \begin{bmatrix} I_{n-r} \\ K \end{bmatrix} = 0. \tag{1}$$

If we introduce new variables for entries in K, namely

$$K = \begin{bmatrix} k_1 & k_{r+1} & \cdots & k_{r(n-r-1)+1} \\ k_2 & k_{r+2} & \cdots & k_{r(n-r-1)+2} \\ \vdots & \vdots & \ddots & \vdots \\ k_r & k_{2r} & \cdots & k_{r(n-r)} \end{bmatrix},$$

where $k_1, \ldots, k_{r(n-r)}$ are new variables, we then obtain a system of $n(n-r)$ bilinear equations in variables $x_1, \ldots, x_n$ and $k_1, \ldots, k_{r(n-r)}$ from (1). In the following discussion, we let $\mathbf{x} = (x_1, \ldots, x_m)$ and $\mathbf{k} = (k_1, \ldots, k_{r(n-r)})$.

Depending on how many column vectors in $\begin{bmatrix} I_{n-k} \\ K \end{bmatrix}$

we use, we can construct $n-r$ bilinear subsystems of (1). Denoting those columns vectors by $\mathbf{k}_1$, $\ldots$, $\mathbf{k}_{n-r}$, we can construct subsystems denoted by $\mathbf{sub}_1, \mathbf{sub}_2, \ldots, \mathbf{sub}_{n-r}$(or **full**) for $r = 1, \ldots, n$ as follows.

$$\underbrace{\left(\sum_{i=1}^{m} x_i M_i\right) \cdot \mathbf{k}_1 = \mathbf{0}}_{\mathbf{sub}_1} \quad \underbrace{\left(\sum_{i=1}^{m} x_i M_i\right) \cdot \begin{pmatrix}\mathbf{k}_1 & \mathbf{k}_2\end{pmatrix} = \mathbf{0}}_{\mathbf{sub}_2}$$

$$\vdots$$

$$\underbrace{\left(\sum_{i=1}^{m} x_i M_i\right) \cdot \begin{pmatrix}\mathbf{k}_1 & \mathbf{k}_2 & \cdots & \mathbf{k}_{n-r}\end{pmatrix} = 0.}_{\mathbf{sub}_{n-r} \text{ or } \mathbf{full}}$$

Depending on the parameters used, using subsystems may be more efficient than using the full system. However, $d_{ff}(\mathbf{sub}_i) \geq d_{ff}(\mathbf{full})$ for $1 \leq i \leq n-r-1$. In [27], this is pointed out and they suggest using subsystems that are determined or over-determined since under-determined subsystems tend to have higher first fall degrees and give spurious solutions.

3.3.1 Complexity

In [16], an upper bound on the degree of regularity (d_{reg}) of a random affine bilinear 0-dimensional system with variable sets $\mathbf{x}$ and $\mathbf{y}$ is given, which is

$$d_{reg} \leq \min(\mathrm{card}(\mathbf{x}), \mathrm{card}(\mathbf{y})) + 1,$$

where $\mathrm{card}(\mathbf{x})$ and $\mathrm{card}(\mathbf{y})$ are cardinalities of the variable sets $\mathbf{x}$ and $\mathbf{y}$. It means if the bilinear polynomial system from the Kipnis-Shamir method is random, we will have $d_{reg} \leq \min(m, r(n-r))$. Later, in [27], Verbel et al. showed this bound was overestimated, and gave a tight bound on the first fall degree of the polynomial system yielded from Kipnis-Shamir method with the following two theorems, and they experimentally showed the solving degree are close to their bound as well.

Theorem 1 ([Theorem 1 [27]]). *The non-trivial syzygies obtained from computing the left kernel of $jac_{\mathbf{k}}(\mathbf{full})$ have degree $r+2$.*

This theorem is derived from analyzing $jac_{\mathbf{k}}(\mathbf{full})$. With simple computation, we have

$$jac_{\mathbf{k}}(\mathbf{full}) = I_{n-r} \otimes jac_{\mathbf{k}}(\mathbf{sub}_1),$$

where $\otimes$ means Kronecker product. Therefore, elements in the left kernel of $jac_{\mathbf{k}}(\mathbf{full})$ and $jac_{\mathbf{k}}(\mathbf{sub}_1)$ have the same degree. $jac_{\mathbf{k}}(\mathbf{sub}_1)$ has size $n \times r$, and has maximal minors of degree r. Therefore, $jac_{\mathbf{k}}(\mathbf{full})$ and $jac_{\mathbf{k}}(\mathbf{sub}_1)$ gives non-trivial syzygies of degree $r+2$.

Theorem 2 ([Theorem 2 [27]]). *When $\binom{r}{d} n > \binom{r}{d+1} m, d+1 \leq n+r$, and $m \leq rn$, from $jac_{\mathbf{x}}(\mathbf{full})$ we can construct non-trivial syzygies of degree $d+2$.*

Let $\left(\sum_{i=1}^{m} x_i M_i\right) = (C^{\dagger}_{n\times(n-r)} || L^{\dagger}_{n\times r})$, where $C^{\dagger}$ and $L^{\dagger}$ are $n \times (n-r)$ and $n \times r$ submatrices of $\left(\sum_{i=1}^{m} x_i M_i\right)$, respectively. Note that all entries of $C^{\dagger}$ and $L^{\dagger}$ are linear polynomials in variables $x_1, \ldots, x_m$. Let $C \in \mathbb{F}^{n(n-r)\times m}$ be the matrix with rows $C_1, \ldots, C_{n(n-r)}$ such that $C_{(i-1)(n-r)+j} = (M_{1,(i,j)}, \ldots, M_{m,(i,j)})$, where $M_{k,(i,j)}$ denotes the (i,j)-th entry of the matrix M_k for $i = 1, \ldots, n,\ j = 1, \ldots, n-r$, and $k = 1, \ldots, m$. Similarly, let $L \in \mathbb{F}^{nr\times m}$ be the matrix with rows $L_1, \ldots, L_{nr}$ such that $L_{(i-1)r+j} = (M_{1,(i,n-r+j)}, \ldots, M_{m,(i,n-r+j)})$, where $M_{k,(i,n-r+j)}$ denotes the $(i, n-r+j)$-th entry of the matrix M_k for $i = 1, \ldots, n,\ j = 1, \ldots, n-r$, and $k = 1, \ldots, m$. Then we have

$$jac_{\mathbf{x}}(\mathbf{full}) = (I_n \otimes K)L + C.$$

Verbel et al. [27] gave a method for constructing the left kernel of $(I_n \otimes K)L + C$ and derived Theorem 2.

As its advantages, the Kipnis-Shamir method can construct a polynomial system more easily compared to minors modeling, subsystems can also be used, and depending on the parameters used, using subsystems can be more efficient than using the full system. However, this method introduces more variables than minors modeling, i.e. variables $k_1, \ldots, k_{r(n-r)}$.

4. Our Proposed Method

In this section, we propose a new method that combines Kipnis-Shamir method and minors modeling.

4.1 The Hybrid Method

The new method can be considered as using a subsystem from the Kipnis-Shamir method and a subsystem from minors modeling.

Let $b_{i,j}$ $(1 \leq i,j \leq n)$ be the (i,j)-th component of $\left(\sum_{i=1}^{m} x_i M_i\right)$ and $\mathbf{b}_1, \ldots, \mathbf{b}_n$ be its row vectors. Namely,

$$\left(\sum_{i=1}^{m} x_i M_i\right) = \begin{matrix}\mathbf{b}_1\\ \mathbf{b}_2\\ \vdots\\ \mathbf{b}_r\\ \mathbf{b}_{r+1}\\ \mathbf{b}_{r+2}\\ \vdots\\ \mathbf{b}_n\end{matrix} \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{r,1} & b_{r,2} & \cdots & b_{r,n} \\ b_{r+1,1} & b_{r+1,2} & \cdots & b_{r+1,n} \\ b_{r+2,1} & b_{r+2,2} & \cdots & b_{r+2,n} \\ \vdots & \vdots & \vdots & \vdots \\ b_{n,1} & b_{n,2} & \cdots & b_{n,n} \end{bmatrix}.$$

In order to make $\left(\sum_{i=1}^{m} x_i M_i\right)$ have rank r, its first r rows $\mathbf{b}_1, \ldots, \mathbf{b}_r$ should be linearly independent, and $\mathbf{b}_1, \ldots, \mathbf{b}_r$ and each $\mathbf{b}_i$ for $i = r+1, \ldots, n$ should be linearly dependent, which give us in total $n-r$ linear relations. We can translate the linear dependence of $\mathbf{b}_1, \ldots, \mathbf{b}_{r+1}$ into either "find y_i such that $\sum_{i=1}^{r} y_i \mathbf{b}_i = \mathbf{b}_{r+1}$" or "$(r+1)\times(r+1)$ minors of the matrix with row vectors $\mathbf{b}_1, \ldots, \mathbf{b}_{r+1}$ vanish." Similarly for linear dependency of $\mathbf{b}_1, \ldots, \mathbf{b}_r, \mathbf{b}_k$, where $r+2 \leq k \leq n$. The approach where new variables y_i are introduced corresponds to the Kipnis-Shamir method. Therefore, we are able

to mix these two methods and obtain a hybrid method. For example, we can use the following strategy:

$$\begin{aligned} &\bullet \text{ Linear dependency of } \mathbf{b}_1, \ldots, \mathbf{b}_{r+1} : \\ &\text{find } y_i \text{ s.t. } \sum_{i=1}^{r} y_i \mathbf{b}_i = \mathbf{b}_{r+1}. \\ &\bullet \text{ Linear dependency of } \mathbf{b}_1, \ldots, \mathbf{b}_r, \mathbf{b}_k : \\ &(r+1) \times (r+1) \text{ minors } = 0. \quad (r+2 \leq k \leq n) \end{aligned} \tag{2}$$

In this example, the upper part is the same with $\mathbf{sub}_1$ in the Kipnis-Shamir method and the lower part is a subsystem from minors modeling. In addition, if the minors derived in the lower half of (2) are not enough, we can always choose another set of $r+1$ row vectors of $\left(\sum_{i=1}^{m} x_i M_i\right)$, generate more minors without introducing any extra variables and add to (2). Therefore, our method manages to avoid introducing many extra variables and saves the trouble of computing many minors.

In total, we have $n-r$ linear dependency relations, which are linear dependency of $\mathbf{b}_1, \ldots, \mathbf{b}_r, \mathbf{b}_i$ for each $\mathbf{b}_i (i = r+1, \ldots, n)$. We can either introduce new variables to mathematically realize these relations or use minors. To clarify the ratio of the Kipnis-Shamir method and minors modeling used in the proposed hybrid method, we introduce a new parameter μ. μ is used to indicate the number of linear dependency relations mathematically realized by the Kipnis-Shamir method. When $\mu = n - r$, all of the linear dependency relations are realized by introducing new variables, and then the hybrid method turns into the Kipnis-Shamir method. When $\mu = 0$, only minors are used to realize those $n - r$ linear dependency relations, and in this case, the hybrid method generates a subsystem of minors modeling.

4.2 Complexity Analysis

As shown in [27] about the Kipnis-Shamir method, the more kernel vectors $\mathbf{k}_1, \ldots, \mathbf{k}_{n-r}$ are used, the lower d_{ff} and d_{sol} will get. Our hybrid method uses a mixture of a subsystem from the Kipnis-Shamir method and a subsystem from minors modeling, which means with μ getting larger, just like the Kipnis-Shamir method, the smaller d_{ff} and d_{sol} will get. It means with μ getting larger, adding a subsystem from minors modeling to a subsystem from the Kipnis-Shamir method will not bring any positive effect. Therefore, we suggest to use an underdetermined subsystem from the Kipnis-Shamir method and a subsystem from minors modeling in our hybrid method, which means choosing μ to be very small, often we choose $\mu = 1$. In this subsection, we use $\mu = 1$ case to analyze the complexity of the hybrid method.

We follow the analysis in [27], that is to study the first fall degree by analyzing non-trivial syzygies. When $\mu = 1$, using the hybrid method, we will obtain a polynomial system given in (2). Note that $\sum_{i=1}^{r} y_i \mathbf{b}_i = \mathbf{b}_{r+1}$ gives a bilinear system in variables $\mathbf{x} = (x_1, \ldots, x_m)$ and $\mathbf{y} = (y_1, \ldots, y_r)$. We denote this polynomial system as S_b. On the other hand, the polynomials in (2) obtained using $(r+1) \times (r+1)$ minors form a quadratic polynomial system in variables $\mathbf{x} = (x_1, \ldots, x_m)$. We denote this polynomial system as S_m. Let $F_{\mu=1} = S_b \cup S_m$, that is, $F_{\mu=1}$ is the system given in (2).

Regarding the first fall degree of a polynomial system, we have the following proposition.

Proposition 2. *Let $F = (f_1, \ldots, f_m) \in R^m$ be a set of polynomials. Then the first fall degree of the subsystems of F bounds the first fall degree of F from above.*

Proof. We will discuss this in two cases according to whether any non-trivial syzygies exist in $\mathcal{I}_R = \langle f_1, \ldots, f_m \rangle$. If F is regular or semi-regular (see [2]), then it does not have non-trivial syzygies, its first fall degree and degree of regularity are the same. Subsystems of F can be regarded as deleting a few polynomials from F, which will increase its degree of regularity or not change it.

If F is not regular nor semi-regular, it has non-trivial syzygies. Suppose we are given a subsystem $(f_1, \ldots, f_s)$ of F, where $s < n$, and it has a non-trivial syzygy $(s_1, \ldots, s_s)$ of degree d. Then we can construct a non-trivial syzygy of degree d of F, which is $(s_1, \ldots, s_s, 0, \ldots, 0)$. According to the definition of the first fall degree, we know the first fall degree of F is at most d since there may exist other non-trivial syzygies of F that have a smaller degree than d. Therefore, the statement is proved. □

According to this proposition, the first fall degrees of S_b and S_m bound the degree of regularity of $F_{\mu=1}$ from above. Since analyzing S_m is difficult, we focus on the bilinear structure of S_b. And we use the kernel of Jacobian matrices of S_b to investigate its first fall degree. The system S_b can be expressed with the following:

$$\underbrace{[y_1 \quad y_2 \quad \cdots \quad y_r \quad 1 \quad 0 \quad \cdots \quad 0] \left(\sum_{i=1}^{m} x_i M_i\right) = 0.}_{S_b} \tag{3}$$

We compute its Jacobian matrices with respect to variables $\mathbf{x}$ and $\mathbf{y}$, respectively.

Jacobian with respect to y

The left kernel of

$$jac_{\mathbf{y}}(S_b)' := \left[\begin{array}{c|c} & b_{1,1} \\ jac_{\mathbf{y}}(S_b) & b_{2,1} \\ & \vdots \\ & b_{n,1} \end{array}\right]$$

gives non-trivial syzygies of S_b, where $b_{i,j}$ is (i,j)-th entry of $\left(\sum_{i=1}^{m} x_i M_i\right)$. Since the kernel of a matrix with linear polynomial entries have elements in the span of its maximal minors, and $jac_{\mathbf{y}}(S_b)'$ is an $n \times (r+1)$ matrix, we know $jac_{\mathbf{y}}(S_b)'$ has maximal minors of degree $r+1$, which give us non-trivial syzygies of degree $r+3$.

Jacobian with respect to x

The left kernel of $jac_{\mathbf{x}}(S_b)$ gives non-trivial syzygies of S_b.

$jac_{\mathbf{x}}(S_b)$ is an $n \times m$ matrix, which has maximal minors of degree m. Therefore, $jac_{\mathbf{x}}(S_b)$ gives us non-trivial syzygies of degree $m+2$.

Combining those two types of non-trivial syzygies, we know the non-trivial syzygies of S_b should have degree

$$\leq \min(r+3, m+2).$$

However, we also have to consider the existence of common divisors among elements in the span of maximal minors of $jac_{\mathbf{y}}(S_b)'$, and $jac_{\mathbf{x}}(S_b)$, respectively. For example, if there exists a common divisor of degree d among elements in the left kernel of $jac_{\mathbf{x}}(S_b)$, then S_b should have non-trivial syzygies of degree less than or equal to $m+2-d$. Experimentally, we found that when $n \geq m+r$ holds, such common divisor appears, which means when $n \geq m+r$, there exists non-trivial syzygies of degree

$$\leq \min(r+2, m+1).$$

Moreover, besides non-trivial syzygies derived from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$, there exists other non-trivial syzygies. Their existence can be confirmed through verifying whether the first fall degree of S_b is lower than the smallest degree of the non-trivial syzygies derived from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$. We conduct some experiments on non-trivial syzygies derived from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$, and its actual first fall degree under different parameter sets. The results are shown in Table 1.

Table 1 Degree of non-trivial syzygies of S_b from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$, its first fall degree and solving degree. In this table, $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$ denote the degrees of the non-trivial syzygies derived from the left kernel of $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$, respectively. d_{ff} and d_{sol} denote the first fall degree and the solving degree of S_b using F4 algorithm implemented in Magma [5], respectively

(q,n,m,r)	$jac_{\mathbf{y}}(S_b)'$	$jac_{\mathbf{x}}(S_b)$	d_{ff}	d_{sol}
(7,8,7,3)	6	9	5	8
(7,9,7,3)	6	9	5	8
(7,10,7,3)	5	9	4	5
(7,11,7,3)	5	8	4	5
(7,12,7,3)	4	6	4	4
(7,7,6,4)	7	8	6	7
(7,8,6,4)	7	8	6	7
(7,9,6,4)	7	8	6	8
(7,10,6,4)	6	8	5	6
(7,11,6,4)	6	6	4	5
(7,12,6,4)	5	7	4	4

This table supports S_b having non-trivial syzygies of degree $\leq \min(r+3, m+2)$. It also shows besides those non-trivial syzygies obtained from Jacobian matrices, there exist other non-trivial syzygies. Despite their existence, and making the first fall degree of S_b lower. To analyze those non-trivial syzygies are very difficult. Considering the existence of common divisors among elements in the span of maximal minors of $jac_{\mathbf{y}}(S_b)'$, and $jac_{\mathbf{x}}(S_b)$, we conclude with the following upper bounds for the first fall degree of S_b :

$$\begin{aligned} n < m+r, &\quad d_{ff} \leq \min(r+3, m+2), \\ n = m+r, &\quad d_{ff} \leq \min(r+2, m+1), \\ n > m+r, &\quad d_{ff} < \min(r+2, m+1). \end{aligned} \tag{4}$$

Finally, since $F_{\mu=1}$ consists of S_b and S_m, there should exist other non-trivial syzygies, and possibly with lower degree than $\min(r+3, m+2)$. This implies that the hybrid method can only bring a positive effect on the Kipnis-Shamir method.

4.3 Further Improvement

In this section, we consider applying the hybrid approach [3] on the hybrid method. That is to exhaustively guess a few variables before applying Gröbner basis computing algorithm on the polynomial system obtained by the hybrid method. The question here is to guess which variables. In both the Kipnis-Shamir method and the hybrid method, we have bilinear systems, which means two sets of different variables. We want to find the set of variables to guess in the hybrid approach that minimizes the total complexity.

One approach we can explore is to utilize the non-trivial syzygies obtained from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$, and see what happens when certain variables are specified. Note that those non-trivial syzygies are in the span of the maximal minors of $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$, and these two Jacobian matrices have entries of linear polynomials. For example, $jac_{\mathbf{x}}(S_b)$ is an $n \times m$ matrix with entries of linear polynomials in variables $y_1, \ldots, y_r$. Its maximal minors should have monomial terms $y_1^m, y_2^m, \ldots, y_r^m$. It implies specifying any variables in $y_1, \ldots, y_r$ will not change the degree of the maximal minors. However, we also have to consider the common divisors among those maximal minors after we specify any variables, which means it is possible to have non-trivial syzygies with a lower degree after all. But analyzing those common divisors can be very difficult. In Table 2, we show the changes on degree of the non-trivial syzygies obtained from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$ after we specify some variables under the parameter set $(q,n,m,r) = (7,13,8,5)$.

Table 2 The changes on degree of non-trivial syzygies obtained from $jac_{\mathbf{y}}(S_b)'$ and $jac_{\mathbf{x}}(S_b)$ under the parameter $(q,n,m,r) = (7,13,8,5)$ when some variables are specified

Number of x variables specified	0	1	2	3	4	5	6	7	8
Degree of syzygies from $jac_{\mathbf{y}}(S_b)'$	7	8	7	6	5	4	4	2	2

Number of y variables specified	0	1	2	3	4	5
Degree of syzygies from $jac_{\mathbf{x}}(S_b)$	10	9	6	5	3	2

This table tells us with more variables specified, the first fall degree of S_b is indeed decreasing. Moreover, when more than 2 variables are specified, specifying variables in $y_1, \ldots, y_r$ brings better results. Since there exist other non-trivial syzygies, first fall degrees can be different from data shown in Table 2. Also since $F_{\mu=1}$ consists of S_b and S_m, specifying variables in $x_1, \ldots, x_m$ may bring decrease in the first fall degree of $F_{\mu=1}$ because of S_m. However, overall these

analyses can be very difficult.

Another approach is through actual experiments. We carry out experiments on different parameter sets with some number of either $\mathbf{x}$ variables or $\mathbf{y}$ variables specified. The results are shown in Table 3.

Table 3 Changes of the first fall degrees and solving degrees under different parameter sets with some variables specified

(q,n,m,r)			# of $\mathbf{x}$ variables fixed						# of $\mathbf{y}$ variables fixed							
			0	1	2	3	4	5	0	1	2	3	4	5	6	7
$(7,10,5,7)$	$\mu=1$	d_{ff}	7	5/6	4	3	2	1	7	7	5/6	4	3	3	2	1
		d_{sol}	9	9	8	3	2	1	9	9	8	4	3	3	2	1
	$\mu=2$	d_{ff}	5	4	3	3	2	1	5	5	4	4	3	3	2	2
		d_{sol}	5	4	4	3	2	1	5	5	5	4	3	3	2	2
	$\mu=3$	d_{ff}	4	4	3	3	2	1	4	4	4	4	3	3	2	1
		d_{sol}	4	4	3	3	2	1	4	4	4	4	3	3	2	1

(q,n,m,r)			# of $\mathbf{x}$ variables fixed								# of $\mathbf{y}$ variables fixed					
			0	1	2	3	4	5	6	7	0	1	2	3	4	5
$(7,10,7,5)$	$\mu=1$	d_{ff}	6	6	5/6	4	3	3	2	1	6	5/6	4	3	2	1
		d_{sol}	6	6	6	4/6	3	3	2	1	6	6	6	3	2	1
	$\mu=2$	d_{ff}	4	4	3/4	3	3	2/3	2	1	4	4	4	3	2	1/2
		d_{sol}	5	4	4	3	3	3	2	1	5	5	4	3	2	1/2
	$\mu=3$	d_{ff}	4	4	3/4	3	3	2/3	2	1	4	4	4	3	2	2
		d_{sol}	4	4	4	3	3	2/3	2	1	4	4	4	3	2	2

Table 3 indicates the set of variables that has smaller cardinality plays a more significant role when we are considering the hybrid approach [3] on the hybrid method. Moreover, specifying each variable from the lower cardinality side approximately decreases the first fall degree by 1, and no matter which μ is used, specifying either m variables in $\mathbf{x}$ or r variables from $\mathbf{y}$ results in having $d_{ff} = d_{sol} = 1$ or 2. We can also see from Table 3 that in the case of $r = 5$, minors have degree 6, which contributes $F_{\mu=1}(7,10,7,5)$ to have $d_{ff} = d_{sol} = 6$, and in the case of $r = 7$, minors have degree 8, which do not contribute to d_{ff} and d_{sol} of $F_{\mu=1}(7,10,5,7)$. What's more, we can notice when the Kipnis-Shamir part is determined or close to be determined, upper bound of d_{ff}, $\min(r+3, m+2)$, can also be applied to d_{sol}. Summarizing the discussion, if we assume using XL algorithm and only multiplying only by monomials from variables $\mathbf{y}$, the complexity of our hybrid method is

$$O\left(q^k \cdot \binom{r\mu - k + d_{ff} - k - 1}{d_{ff} - k}^{\omega}\right),$$

where d_{ff} has upper bound given in Eq (4), and $2 \le \omega \le 3$ is the linear algebra constant. Note that this complexity works very well with finite fields of small cardinalities.

5. Experiments and Application

5.1 Experiments

The parameters we choose to run experiments on proportionally coincide with Rainbow [12], which are $(q, 3a, 2a, 2a)$ and $(q, 4a, 2a, 3a)$, where $a = 2, 3$ and q is the cardinality of the finite field used. We run experiments 5 times for each set of parameter.

Table 4 presents results on minors modeling, T_{minors} represents the time cost for constructing all minors needed, T_{F4} represents the time cost for solving the minrank problem with those minors using F4 algorithm implemented in Magma, and $T_{\text{minors}} + T_{\text{F4}}$ represents the total time cost of solving the minrank problem.

Table 5 presents results on the Kipnis-Shamir method, and $\mathbf{sub}_2, \mathbf{sub}_3$ represent solving the minrank problem using subsystems of the Kipnis-Shamir system derived from using $2, 3$ kernel vectors, T_{F4} represents the time cost for solving the minrank problem with F4 algorithm in Magma.

Table 6 gives experimental results on the proposed hybrid method, T_{minors} represents the time cost for constructing minors needed, and T_{F4} represents the time cost for solving the polynomial system obtained in the hybrid method. The total time cost for the hybrid method is $T_{\text{minors}} + T_{\text{F4}}$. Note when $(q, n, m, r, \mu) = (16, 9, 6, 6, 1)$ and $(16, 12, 6, 9, 1)$, our hybrid method outperforms both the Kipnis-Shamir method and minors modeling, and they are marked in bold.

Table 7 shows the behavior of d_{ff} and d_{sol} when a certain amount of $\mathbf{x}$ variables are specified.

All of our experiments were executed on a 2.10 GHz Intel® Xero® Gold 6130 Processor with Magma V2.24-8 [5], where F4 algorithm [14] is implemented. From tables 4, 5, 6, we know under the proportioned Rainbow secure parameters, our hybrid method ($\mu = 1$) solves the Minrank problem faster than the Kipnis-Shamir method, and requires way less minors than the minors modeling.

Table 4 Experimental results on solving minrank instances with minors modeling in shown §3.2

(q,n,m,r)	d_{ff}	d_{sol}	T_{minors} $[s]$	T_{F4} $[s]$	$T_{\text{minors}} + T_{\text{F4}}$ $[s]$
$(16,8,4,6)$	13	13	0.47	0.05	0.52
$(16,9,6,6)$	7	7	39.20	0.98	40.18
$(16,12,6,9)$	10	10	5992.43	32.82	6025.25

Table 5 Results on solving minrank instances with the Kipnis-Shamir method shown in §3.3

(q,n,m,r)		d_{ff}	d_{sol}	T_{F4} $[s]$
$(16,8,4,6)$	$\mathbf{sub}_2$	4	5	3.54
$(16,9,6,6)$	$\mathbf{sub}_2$	5	6	255.62
	$\mathbf{sub}_3$	4	5	128.68
$(16,12,6,9)$	$\mathbf{sub}_2$	6	6	109361.56
	$\mathbf{sub}_3$	5	6	> 172800

Table 6 Results on solving minrank instance with our hybrid method given in §4.1

(q,n,m,r)		d_{ff}	d_{sol}	T_{minors} $[s]$	T_{F4} $[s]$	$T_{\text{minors}} + T_{\text{F4}}$ $[s]$
$(16,8,4,6)$	$\mu=1$	6	8	0.24	0.78	1.02
	$\mu=2$	4	5	0.18	3.71	3.89
$(16,9,6,6)$	$\mu=\mathbf{1}$	8	8	**6.53**	**7.74**	**14.27**
	$\mu=2$	5	7	5.44	243.15	248.59
	$\mu=3$	4	5	4.36	118.07	122.43
$(16,12,6,9)$	$\mu=\mathbf{1}$	8	11	**544.77**	**1945.75**	**2490.52**
	$\mu=2$	6	6	453.97	> 172800	> 173253.97

Table 7 Results on solving minrank instances using hybrid method with hybrid approach of polynomial solving, note that only variables $x_1, \ldots, x_m$ are specified

(q,n,m,r)			Number of x variables fixed								
			0	1	2	3	4	5	6	7	8
$(16,8,4,6)$	$\mu=1$	d_{ff}	6	4/5	3	2	1	-	-	-	-
		d_{sol}	8	8	7	2	1	-	-	-	-
	$\mu=2$	d_{ff}	4	3	3	2	1	-	-	-	-
		d_{sol}	5	4	3	2	1	-	-	-	-
$(16,9,6,6)$	$\mu=1$	d_{ff}	8	7	5/6	4	3	2	1	-	-
		d_{sol}	8	8	8	7	3	2	1	-	-
	$\mu=2$	d_{ff}	5	5	4	3	3	2	1	-	-
		d_{sol}	6	5	4	3	3	2	1	-	-
$(16,12,6,9)$	$\mu=1$	d_{ff}	8	7	5/6	4	3	2	1	-	-
		d_{sol}	11	10	10	10	3	2	1	-	-
	$\mu=2$	d_{ff}	6	5	4	4	3	2	1	-	-
		d_{sol}	6	5	5	4	3	2	1	-	-

5.2 Application on MPKC

Rainbow A public key from Rainbow(q, v, o_1, o_2) can give us a minrank instance $(q, v + o_1 + o_2, o_1 + o_2, v + o_1)$. For example Rainbow$(16, 32, 32, 32)$, which achieves NIST type I security, gives us a minrank instance $(16, 96, 64, 64)$. If we use minors method, d_{reg} is estimated to be 65, assume $\omega = 2.4$ then we have complexity 2^{297}. If we use Kipnis-Shamir full system, d_{reg} is estimated to be 29, if we assume using 5 out of 32 kernel vectors and $d_{reg} = 29$, the estimated complexity is then 2^{337}. If we use hybrid method $\mu = 1$ with hybrid approach of polynomial solving, with 63 $\mathbf{y}$ variables specified, we have $d_{reg} = 2$, and the complexity is 2^{279}.

6. Conclusion

In this paper, methods for solving the minrank problem are considered. We reviewed two of the existing methods, the Kipnis-Shamir method, and minors modeling, and some results on their complexities. We proposed a hybrid method that combined Kipnis-Shamir method and minors modeling. Different from Kipnis-Shamir method, this new method manages to avoid introducing many variables, and unlike minors modeling, it does not require computation of many minors. Our hybrid method solves the minrank problem by solving a polynomial system that consists of a subsystem from the Kipnis-Shamir method and a subsystem from minors modeling. Then the subsystem from the Kipnis-Shamir method is a bilinear polynomial system.

Moreover, we consider applying the hybrid approach of solving multivariate polynomials on our hybrid method, which means a few variables are specified before solving the polynomial system obtained from our hybrid method. Since the polynomial system obtained from our hybrid method has a bilinear subsystem, we considered the significance of specifying each set of variables. We experimentally verified specifying the variables that have smaller cardinality brings better results. Finally, we applied our hybrid method with some variables specified to Rainbow and verified its effectiveness.

References

[1] Gorjan Alagic, Jacob Alperin-Sheriff, Daniel Apon, David Cooper, Quynh Dang, Yi-Kai Liu, Carl Miller, Dustin Moody, Rene Peralta, Ray Perlner, Angela Robinson, and Daniel Smith-Tone. Status report on the first round of the nist post-quantum cryptography standardization process. NIST Internal Report 8240, National Institute of Standards and Technology, 2018.

[2] M. Bardet, J. C. Faugère, B. Salvy, and B. Y. Yang. Asymptotic behavior of the index of regularity of quadratic semi-regular polynomial systems. In *8th International Symposium on Effective Methods in Algebraic Geometry –MEGA'05*, 2005.

[3] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Hybrid approach for solving multivariate systems over finite fields. *Journal of Mathematical Cryptology*, 3:177–197, 2009.

[4] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Cryptanalysis of hfe, multi-hfe and variants for odd and even characteristic. *Designs, Codes and Cryptography*, 69(1):1–52, 2013.

[5] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997.

[6] B. Buchberger. *Ein Algorithmus zum Auffinden der Basiselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*. PhD thesis, Universitat Innsbruck, 1965.

[7] Jonathan F. Buss, Gudmund S. Frandsen, and Jeffrey O. Shallit. The computational complexity of some problems of linear algebra. *Journal of Computer and System Sciences*, 58(3):572 – 596, 1999.

[8] Cabarcas, Daniel Smith-Tone, and Javier A. Verbel. Key recovery attack for ZHFE. In *Post-Quantum Cryptography 2017*, volume 10346 of *LNCS*, pages 289–308. Springer, 2017.

[9] Alessio Caminata and Elisa Gorla. The complexity of minrank. arXiv:1905.02682 [cs.SC], 2019.

[10] Lily Chen, Stephen Jordan, Yi-Kai Liu, Dustin Moody, Rene Peralta, Ray Perlner, and Daniel Smith-Tone. Report on post-quantum cryptography. NIST Interagency Report 8105, National Institute of Standards and Technology, 2016.

[11] Nicolas Courtois, Alexander Klimov, Jacques Patarin, and Adi Shamir. Efficient algorithms for solving overdefined systems of multivariate polynomial equations. In *Advances in Cryptology – EUROCRYPT 2000*, volume 1807 of *LNCS*, pages 392–407. Springer, 2000.

[12] Jintai Ding and Dieter Schmidt. Rainbow, a new multivariate polynomial signature scheme. In *Applied Cryptography and Network Security – ACNS 2005*, volume 3531 of *LNCS*, pages 164–175. Springer, 2005.

[13] Jean-Charles Faugeère, Mohab Safey El Din, and Pierre-Jean Spaenlehauer. Computing loci of rank defects of linear matrices using grÖbner bases and applications to cryptology. In *Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation*, ISSAC '10, pages 257–264. ACM, 2010.

[14] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases (F4). *Journal of Pure and Applied Algebra*, 139(1):61 – 88, 1999.

[15] Jean Charles Faugère. A new efficient algorithm for computing Gröbner Bases without reduction to zero (F5). In *ISSAC 2002*, pages 75–83. ACM, 2002.

[16] Jean-Charles Faugère, Mohab Safey El Din, and Pierre-Jean Spaenlehauer. Gröbner bases of bihomogeneous ideals generated by polynomials of bidegree (1,1): Algorithms and complexity. *Journal of Symbolic Computation*, 46(4):406 – 437, 2011.

[17] Jean-Charles Faugère, Mohab Safey El Din, and Pierre-Jean Spaenlehauer. On the complexity of the generalized minrank problem. *Journal of Symbolic Computation*, 55:30 – 58, 2013.

[18] Louis Goubin and Nicolas T. Courtois. Cryptanalysis of the ttm cryptosystem. In *Advances in Cryptology – ASIACRYPT 2000*, volume 1976 of *LNCS*, pages 44–57. Springer, 2000.

[19] Aviad Kipnis and Adi Shamir. Cryptanalysis of the HFE public key cryptosystem by relinearization. In *Advances in Cryptology – CRYPTO' 99*, volume 1666 of *LNCS*, pages 19–30. Springer, 1999.

[20] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209, 1987.

[21] Jacques Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. In *Advances in Cryptology – EUROCRYPT '96*, volume 1070 of *LNCS*, pages 33–48. Springer, 1996.

[22] Jacques Patarin, Nicolas Courtois, and Louis Goubin. QUARTZ, 128-bit long digital signatures. In *Topics in Cryptology – CT-RSA 2001*, volume 2020 of *LNCS*, pages 282–297. Springer, 2001.

[23] Ray Perlner, Albrecht Petzoldt, and Daniel Smith-Tone. Total break of the srp encryption scheme. In *Selected Areas in Cryptography – SAC 2017*, LNCS, pages 355–373. Springer, 2018.

[24] Albrecht Petzoldt, Ming-Shing Chen, Bo-Yin Yang, Chengdong Tao, and Jintai Ding. Design principles for HFEv-based multivariate siganture schemes. In *Advances in Cryptology – ASIACRYPT 2015*, volume 9452 of *LNCS*, pages 311–334. Springer, 2015.

[25] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126, 1978.

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

[27] Javier Verbel, John Baena, Daniel Cabarcas, Ray Perlner, and Daniel Smith-Tone. On the complexity of "superdetermined" minrank instances. Cryptology ePrint Archive, Report 2019/731, 2019. .