

博士論文

Security Evaluation of
Multivariate Public-Key Cryptography
via Algebraic Techniques
(代数的手法による多変数多項式暗号
の安全性評価)

王 亜成

Abstract

Public-key cryptography, as one of the most fundamental security tools in modern technology, assures secure communications and data storage on the internet. Three different applications of public-key cryptography include encryption, digital signatures and key establishment. Up until now, RSA cryptosystem and elliptic curve cryptosystem, which are respectively based on large integer factorization problem and discrete logarithm problem, are widely used in every corner of our everyday use. With the advent of new technologies, quantum computing, their security is at stake as quantum algorithms for solving the aforementioned problems are proposed already, which leads us to a new era of cryptography called post-quantum cryptography.

Post-quantum cryptography (PQC) refers to cryptosystems that are based on hard problems for solving even with the help of quantum computers. Since 2016, a standardization for PQC has been carried out by National Institute of Standards and Technology and more studies on PQC has emerged. Besides other candidates such as lattice-based cryptography, code-based cryptography, hash-based cryptography and isogeny-based cryptography, multivariate public-key cryptography also is expected to form some good candidates for PQC.

Multivariate public-key cryptography (MPKC) has a set of multivariate quadratic polynomials as its public key, which quantum computers have no advantages in solving. Those polynomials in public keys often result in very large keys for MPKC, it is one of the research topics for MPKC. In addition, constructing random enough polynomials that can be used for public keys, and thorough security analysis are also often discussed topics.

In this thesis, we focus on the algebraic aspect of MPKC's security and our topics will be orbiting around polynomial solving. In general, there are three paths for cryptanalysis of MPKC via algebraic approach. Most intuitively, we solve a polynomial system from a public key by computing its Gröbner bases directly. This approach can possibly be improved by coupling Gröbner bases computation with exhaustive search, and we want to find a best strategy for this direct solving. Since public keys of all multivariate cryptosystems are constructed from some special polynomials, which enables efficient inversion, this often leads to hidden structures in a public key. For example, minrank

attack and rainbow band separation attack against some multivariate cryptosystems fall into this category. Those hidden structures allow transforming a public key into a new set of polynomials with different property. Then we use direct solving again. Finally, polynomial systems obtained from a multivariate cryptosystem are all over finite fields. By exploring relations with subfields and using trivial relations on variables over a finite field, we are able to transform a polynomial system into a new polynomial system over a subfield with additional relations on variables, then we evaluate its solving complexity. These three algebraic approaches help us bring new sights to the security of MPKC.

Keywords Multivariate public-key cryptography, Post-quantum cryptography, Gröbner basis, Hybrid approach, Degree of regularity, First fall degree

Contents

Chapter 1	Introduction	1
1.1	Background	4
1.2	Motivation	5
1.3	Contribution	11
1.4	Organization	14
Chapter 2	Multivariate public-key cryptography	16
2.1	Multivariate quadratic problem	18
2.2	The Rainbow signature scheme	21
2.3	The HFEv- signature scheme	42
2.4	Classification on MPKC	50
Chapter 3	Algebraic techniques for solving polynomials	51
3.1	Rings and Ideals	52
3.2	Commutative algebra and algebraic geometry	55
3.3	Gröbner bases and algorithms	66
3.4	Solving polynomial systems	77
Chapter 4	Evaluating the security of EFC using algebraic techniques	84
4.1	Introduction	85
4.2	Multivariate cryptography	86
4.3	Extension field cancellation (EFC)	88
4.4	Proposed efficient decryption algorithms for EFC	94
4.5	Hybrid attack against EFC	99
4.6	Conclusion	104

Chapter 5	On the algebraic aspects of solving the minrank problem	105
5.1	Introduction	106
5.2	The MQ problem and bilinear systems	108
5.3	The minrank problem	111
5.4	Our proposed method	113
5.5	Experiments and application	118
5.6	Conclusion	122
Chapter 6	Algebraic cryptanalysis of multivariate encryption scheme PERN	124
6.1	Introduction	124
6.2	Multivariate encryption scheme PERN	126
6.3	Algebraic cryptanalysis of PERN	133
6.4	Secure parameters	136
6.5	Conclusion	136
Chapter 7	On the Weil descent attack against the MQ problem	138
7.1	Introduction	139
7.2	The multivariate quadratic problem	141
7.3	Weil descent on the MQ problem	143
7.4	Experiments and comparison	150
7.5	Conclusion	154
Chapter 8	Conclusion and future work	155
8.1	Conclusion	155
8.2	Future work	157
	Acknowledgements	159
	Bibliography	161

Chapter 1

Introduction

The most important aspect involving informational communication is security, and for most modern telecommunication technologies such as SSH, SSL/TLS and IPsec, public-key cryptography is applied to achieve the required level of security. Up until now, public-key cryptosystems RSA [RSA78] and ECC [Kob87], which are respectively based on large integer factorization problem and discrete logarithm problem, have been used. However, everything has changed since Shor [Sho97] proposed quantum polynomial-time algorithms for solving those two problems in 1994, which poses a major threat on modern communication security. Developing quantum resistant cryptographic algorithms hence becomes an urgent task. This new quantum resistant cryptography is called post-quantum cryptography (PQC).

Ever since National Institute of Standards and Technology (NIST) [CJL⁺16] in the United States started the project of standardizing post-quantum cryptography in 2016, more attention on this field has been paid. This standardization project by NIST calls for proposals for digital signatures, public-key encryption, and key-establishment algorithms. Up until its first round submission deadline in November, 2017, 69 candidate algorithms met the minimum acceptance criteria and submission requirements, and were published for further analysis. Those candidates contain lattice-based cryptosystems, code-based cryptosystems, multivariate cryptosystems, hash-based signatures and isogeny-based cryptosystems. Fast forward to January 2019, second round candidates were announced and 26 algorithms out of 69 candidates in the first round were selected. More recently, in July 2020, third round finalists were announced to be considered for standardization at the end of the third round. For public-key encryption and key-establishment algorithms, Classic McEliece, CRYSTALS-KYBER, NTRU, and

SABER were selected, and for digital signatures, CRYSTALS-DILITHIUM, FALCON, and Rainbow were selected. In addition to the chosen finalists, eight alternate candidate algorithms were also chosen to advance to the third round, which are BIKE, FrodoKEM, HQC, NTRU Prime, SIKE, GeMSS, Picnic, and SPHINCS+.

Multivariate public-key cryptography (MPKC) refers to public-key cryptography that uses a set of multivariate non-linear polynomials over a finite field as its public key, its security is based on the hardness of solving this polynomial system. More specifically, when those non-linear polynomials are quadratic, the problem of solving this polynomial system is called the multivariate quadratic problem (MQ problem), which is proven to be NP-complete [RGSJ79] when the finite field is the field of order 2. For better understanding of the hardness of the MQ problem, from April 2015, a contest, Fukuoka MQ challenge [YDH⁺15b], dedicated to solving polynomial systems over finite fields started. Besides considering the attack of solving the polynomial system given by a public key, some structural attacks against multivariate cryptosystems should also be considered since some special structures can be hidden in the public key polynomials due to its key construction.

Attacks on multivariate cryptosystems can be divided into two categories, algebraic attacks and structural attacks. Algebraic attacks solve the polynomial system derived from a ciphertext and a public key by computing a Gröbner basis using linearization techniques [CKPS00] algorithm or Gröbner basis techniques [Buc65]. Gröbner basis techniques are commonly used efficient tools for attacking multivariate cryptosystems. Since the notion was proposed, many algorithms for computing a Gröbner basis had been proposed, for example Buchberger algorithm [Buc65], and its improved version using linear algebra techniques by Frauère, F4 [Fau99] and F5 [Fau02]. There are also some research results published on the relation between extended linearization algorithm (XL) and F4 algorithm [AFI⁺04], and it shows XL algorithm is a redundant version of F4 algorithm. On the other hand, structural attacks exploit the special structure in the key construction of a multivariate cryptosystem and find its weakness, and subsequently apply special attacks on it. Typical attacks that fall into this category include rank attacks [CSV93a, GC00a], linearization attack [Pat95], differential attack [DFSS07, FGS05]. Parameter selection and security evaluation of a multivariate cryptosystem is done via considering the most efficient applicable attacks on it.

Compared to other candidates for post-quantum cryptography, multivariate public-key

cryptography has advantages such as very short digital signature, modest computational resources requirements, high speed performance, etc. These advantages are mainly due to the usage of multivariate quadratic polynomials over finite fields of small orders and most computations can be accomplished using matrices. Digital signatures schemes Rainbow [DS05] and GeMSS [CFMR⁺17] are typical multivariate cryptosystems that have very short signature length, which is also one of the reasons they are kept in the third round finalist in NIST post-quantum standardization project. Conversely, multivariate public-key cryptography also has drawbacks such as very large public key size, difficulties in building secure public-key encryption and lacking security proofs, etc. Large public key size is due to using many multivariate quadratic polynomials in a public key. Therefore, thorough security analysis, public key size reduction and constructing secure public-key encryption are the main topics in multivariate cryptography.

In modern cryptographic technologies, most widely used algorithms include Diffie-Hellman key exchange [DH76], RSA encryption and signature [RSA78] and elliptic curve cryptosystem [Kob87]. They are based on large integer factorization problem or discrete logarithm problem, which are hard problems for classical computation resources and there only exist sub-exponential time algorithms on classical computers for solving them. In 1994, Shor proposed polynomial-time algorithms on quantum computers for solving those problems, which means with advent of large scale quantum computers current cryptographic technologies collapse. According to a report [CJL⁺16] on post-quantum cryptography by NIST, quantum computers that are capable enough of breaking 2048 bit RSA cryptosystem are expected to be built in around 2030. Besides the threat posed by Shor's algorithm, Grover's quantum search algorithm also affects many cryptosystems. Grover's quantum search algorithm finds a unique data in an unsorted database with N entries in $\mathcal{O}(\sqrt{N})$ time. Due to this algorithm, symmetric key cryptosystems need to increase their key size to double, and it also affects public key cryptosystems when attack of exhaustive search is applied. In table 1.1, impact of quantum computers on currently used symmetric key cryptosystem AES and public-key cryptosystems RSA and ECC is presented.

In August 2015, National Security Agency of United States of America announced their plan of transition to post-quantum cryptography in the near future, which drew a lot of attention on post-quantum cryptography. Moreover, in February 2016 at the conference PQCrypto 2016, NIST published their call for submissions for standardiz-

Table 1.1: Impact of quantum computers on currently used cryptosystems

Cryptosystems	Type	Function	Impact
AES	Symmetric key	Encryption	Key size doubled
RSA	Public key	Encryption and Signature	Broken
ECC	Public key	Key exchange and Signature	Broken

ing post-quantum cryptography. With those actions been taken, many workshops and conferences on post-quantum cryptography have been held.

1.1 Background

According to a NIST report [CJL⁺16] on post-quantum cryptography, candidates like lattice-based cryptography, code-based cryptography, multivariate public-key cryptography, hash-based signatures and isogeny-based cryptography are considered. NIST plans to standardize public-key encryption, digital signature and key exchange. Since the initial announcement in February 2016, many actions have been taken on this project. Table 1.2 lists all the recent movements related to this standardization project.

For public-key encryption, NIST requires submissions to possess indistinguishability under adaptive chosen ciphertext attack, which is also denoted IND-CCA2 in the cryptography world. For digital signature, NIST requires submissions to enable existentially unforgeable digital signatures with respect to an adaptive chosen message attack, which is also commonly referred as EUF-CMA security in cryptography. NIST also defined a series of security categories by using reference primitive such as symmetric key encryption scheme AES and hash algorithm SHA. Table 1.3 lists those security categories.

In December 2017, NIST announced 69 submitted algorithms as candidates and 7 of them are multivariate cryptosystems, which are listed in Table 1.4. In January 2019, 26 second round candidates were announced and 4 multivariate signature schemes were chosen, Table 1.5 presents those schemes. Fast forward to July 2020, NIST selected 7 finalists and 8 alternates to move on to the third round of screening, and 1 of those finalists, Rainbow, and 1 of those alternates, GeMSS, are multivariate cryptosystems as shown in Table 1.6.

Table 1.2: Recent movements related to NIST PQC standardization project [AASA⁺20]

Apr. 2015	NIST Workshop on Cybersecurity in a Post-Quantum World
Feb. 2016	Announcement of NIST PQC Standardization Project
Apr. 2016	NISTIR 8105 [CJL ⁺ 16], Report on Post-Quantum Cryptography
Nov. 2017	Submission Deadline for NIST PQC Standardization Project
Dec. 2017	First-round candidates announced
Apr. 2018	First NIST PQC Standardization Conference
Jan. 2019	Second-round candidates announced. NISTIR 8240, Status Report on the First Round of the NIST Post-Quantum Cryptography Standardization Process [AASA ⁺ 18], released.
Apr. 2019	Deadline for updated submissions for the second round
Aug. 2019	Second NIST PQC Standardization Conference
Jul. 2020	Third round finalists and alternate candidates announced. NISTIR 8309 [AASA ⁺ 20], Status Report on the Second Round of the NIST Post-Quantum Cryptography Standardization Process, released.

Table 1.3: Security categories defined by NIST

Category	As secure as	Attack used
1	AES-128	exhaustive key search
2	SHA-256	collision search
3	AES-192	exhaustive key search
4	SHA-384	collision search
5	AES-256	exhaustive key search

1.2 Motivation

Polynomial system solving has always been an important field of study in many fields of engineering and computer science such as cryptography, coding theory, optimization, game theory, statistics, machine learning and numerous other fields. Many hard problems can also be reduced to the problem of solving polynomial systems. In this thesis, we focus

Table 1.4: Multivariate cryptosystems in NIST 1st round candidates

LUOV [WPSV17]
Rainbow [DS05, DCP ⁺ 17b]
HiMQ-3 [SPKK17]
Gui [PCY ⁺ 15, DCP ⁺ 17a]
GeMSS [CFMR ⁺ 17]
DualModeMS [FPR17]
MQDSS [CHR ⁺ 16, CHR ⁺ 17]

Table 1.5: Multivariate cryptosystems in NIST 2nd round candidates

LUOV [WPSV19]
Rainbow [DS05, DCP ⁺ 19]
GeMSS [CFMR ⁺ 19]
MQDSS [CHR ⁺ 16, CHR ⁺ 19]

Table 1.6: Multivariate cryptosystems in NIST 3rd round candidates

Finalist	Rainbow [DCP ⁺ 20]
Alternate	GeMSS [CFMR ⁺ 20]

on multivariate polynomial systems appeared in multivariate public-key cryptography. There are many methods proposed for solving polynomial systems, including numerical methods, multivariate resultants [CD05] and many other methods. In this thesis, we focus on Gröbner basis [Buc65], which is especially effective for polynomial systems over finite fields.

Multivariate public-key cryptography refers to public-key cryptosystems whose public keys are a set of multivariate nonlinear (quadratic) polynomials. Its security is closely related to the problem of solving polynomial systems. The public key polynomial systems are often constructed from a structured polynomial system and two affine transformations. One of the transformations is a change of variables and the other one mixes all polynomials, which result in a randomness in the composition of the three. The most desired cryptosystem designs are those with fully random public key polynomial system,

but it is not an easy task since it is built from a structured polynomial system. Therefore, the goal in the field of multivariate public-key cryptography is building a cryptosystem with public key polynomial system that behaves like random algebraically, where no structural attacks can be applied.

Algebraic cryptanalysis of multivariate public-key cryptography often centers around two directions. One is solve the polynomial system derived from a public key algebraically and analyze its complexity. The other one is use the construction of a multivariate public-key cryptosystem to derive other different polynomial systems, solve them algebraically and analyze their complexities. Algebraic cryptanalysis is essential in multivariate public-key cryptography as it can be applied to all cryptosystems, and it is the most effective tool to analyze its security. Moreover, by comparing the algebraic behavior of a cryptosystem with a random system of the same size, hidden structures of the cryptosystem can sometimes be found, which makes it an essential tool in multivariate public-key cryptography.

When a solving strategy is determined, complexity analysis is the most important in algebraic cryptanalysis. When using algebraic techniques like Gröbner basis [Buc65, Fau99, Fau02] or XL algorithm [CKPS00], their complexity is very closely related to the so-called *degree of regularity*, which is related to the properties of the ideal generated by the given polynomial. For randomly chosen polynomial systems this value is well understood and can be estimated tightly since they are expected to be regular or semi-regular polynomial sequences. For regular or semi-regular polynomial sequences there are no degree falls while computing their Gröbner bases, and their degree of regularity bounds the maximum degree, which makes complexity estimation relatively easy. Conversely, the complexity of the algebraic attack on nonregular or nonsemiregular polynomial sequences needs more insights and analysis. For example, to predict the *degree falls* during the computation of a Gröbner bases.

1.2.1 Cryptanalysis of multivariate cryptosystems via algebraic attacks

For any multivariate public-key cryptosystem, we can obtain a polynomial system from its public key. For a multivariate public-key cryptosystem with a conventional construction, its security is based on the hardness of solving the polynomial system from its public key and finding a decomposition of its public key polynomials into a composition of a set of easy-to-solve polynomials with two affine transformations. The

former is called algebraic attack and the latter is called key recovery attack. For the algebraic attack, Gröbner basis techniques are available. More specifically, algorithms such as F4 [Fau99], F5 [Fau02] and XL [CKPS00] algorithms are available. F4 and F5 algorithms are improved algorithms from Buchberger's algorithm [Buc65], which recursively computes polynomial reductions of the S -polynomials of the pairs in the given polynomial sequence. If the result of the reduction is not zero, it adds the results to the polynomial sequence and repeat the same process until all S -polynomial are reduced to zero. F4 and F5 use linear algebra to carry out polynomial reduction of many pairs at the same time, which enables more efficient computation of Gröbner bases and more concrete complexity estimation. For regular or semi-regular sequence, the degrees of polynomials appeared during the computation of Gröbner bases rise at each step until it reaches a maximum, which is bounded by the degree of regularity and it can be estimated. Fröberg's conjecture states that all randomly chosen polynomial systems are regular or semi-regular systems. For polynomial systems that are not regular or semi-regular, the complexity of computing a Gröbner basis for them is intractable since their degree of regularity have to be analyzed. Considering the public key polynomial systems in multivariate public-key cryptography are all generated from a composition of an easy-to-solve polynomial system and affine maps. Under many circumstances, they are not random. Their algebraic cryptanalysis has to be analyzed individually. If the degree of regularity of a polynomial system is known, say d_{reg} , its complexity of running F4 and F5 algorithm is bounded by

$$O\left(m \binom{n + d_{reg} - 1}{d_{reg}}^\omega\right),$$

where m is the number of polynomials, n is the number of variables and $2 < \omega \leq 3$ is the linear algebra constant.

When considering the algebraic attack against a multivariate public-key cryptosystem, a few techniques can be applied, which are adding trivial field equations, hybrid approach of Gröbner basis techniques and exhaustive search, and making use of the extra variables of an underdetermined polynomial system.

Field equations When solving a polynomial system of m polynomial in n variables over a finite field \mathbb{F}_q , variables satisfy the following trivial equations called *field*

equations:

$$\begin{cases} x_1^q - x_1 = 0 \\ \vdots \\ x_n^q - x_n = 0. \end{cases} \quad (1.1)$$

Theoretically, during the computation of a Gröbner basis for this polynomial system, the maximal polynomial degree appeared is bounded by its degree of regularity (d_{reg}). If $q \leq d_{reg}$, adding those field can bring a positive result.

Hybrid approach Gröbner basis techniques are often coupled with exhaustive search to bring a best performance, this technique is called hybrid approach. More specifically, this approach specifies a few variables in a polynomial system before applying Gröbner basis techniques. If a correct solution is not obtained, it repeats the same process. Hybrid approach is a simple but very effective technique in solving polynomial systems.

Extra variables in an underdetermined system Public keys of multivariate signature schemes are all determined or underdetermined polynomial systems. Random underdetermined polynomial systems have multiple solutions. In cryptographic attacks, only one solution is wanted, hence extra variables are usually specified with random value to fasten the process of computing a Gröbner basis. However, this is known as the worst case complexity since the extra variables are not used. In [Tho13], an algorithm for transforming a polynomial system with m polynomials and n variables into a polynomial system with $m - \lfloor \frac{n}{m} \rfloor - 1$ polynomials and $m - \lfloor \frac{n}{m} \rfloor - 1$ variables is proposed. Therefore, obtaining a solution of an underdetermined polynomial system of m polynomials in n variables is equivalent to solving a polynomial system of $m - \lfloor \frac{n}{m} \rfloor - 1$ polynomials and $m - \lfloor \frac{n}{m} \rfloor - 1$ variables.

Therefore, algebraic cryptanalysis of multivariate public-key cryptography have always been a very important topic in the field of public-key cryptography.

1.2.2 Algebraic cryptanalysis of multivariate cryptosystems with structural transformation

Another direction of algebraic cryptanalysis of multivariate cryptosystems is through structural transformation, which is related to the construction of a cryptosystem. In multivariate public-key cryptography, to construct a cryptosystem, it first chooses an easy-to-invert multivariate nonlinear polynomial map F , then choose two affine maps S, T and a public key is given by a composition of these three maps. One of the affine maps, say S , serves as a change of variables on F , and the other map T mixes polynomials in F . As the map F has to be easy-to-invert, special structures have to be used, which sometimes leads to vulnerabilities of the cryptosystems. For example, when F is chosen to be a set of multivariate quadratic polynomials, if matrices associated to the quadratic forms of the polynomials in F are non-full rank, by finding a linear combination of public key polynomial of low rank, partial information about T can be recovered. This is known as the minrank attack. Once breaking a cryptosystem is modeled as solving the minrank problem, a new polynomial system is often generated to solve this minrank problem. This resulting polynomial system is not random as it comes from a structure, and its solving complexity has to be analyzed to understand the security of a cryptosystem against the minrank attack. When considering solving a polynomial system, besides analyzing its degree of regularity, using different mathematical models are also important. Different models can result significantly different results. For example, when solving the minrank problem, minors modeling and Kipnis-Shamir method can be used. Both of these methods transform solving the minrank problem into the problem of solving a polynomial system, but the polynomial system from the former is related to determinantal polynomials and the latter is related to multi-homogeneous polynomials. On the other hand, there exists equivalent keys in multivariate public-key cryptography. Suppose a public key of a cryptosystem is generated by a composition of T, F, S , namely

$$P = T \circ F \circ S,$$

its equivalent keys T', F', S' satisfy the following two conditions

- $P = T \circ F \circ S = T' \circ F' \circ S,$
- F' has the same structure with F that enables inversion.

By regarding unknowns in equivalent keys T', S' and taking the structure of F' into consideration, a multivariate polynomial system can be formed. To understand the security of a cryptosystem, the complexity of solving the polynomial system derived from equivalent keys has to be analyzed.

1.2.3 More techniques on polynomial solving

Besides hybrid approach and considering extra variables in an underdetermined polynomial system, there are many techniques and algorithms proposed dedicated to solving various different special polynomial systems such as bi-homogeneous polynomial system, determinantal polynomial system, sparse polynomial system and numerous other polynomial systems. For example, to solve the discrete logarithm problem on algebraic curve over composite fields, a technique called Weil descent is used, it transforms a polynomial over a finite field a polynomial system over its subfield. This technique can also be applied to algebraic cryptanalysis of multivariate public-key cryptography, but its complexity analysis is missing.

To summary, my research is centered (see Figure 1.1) around three topics in algebraic cryptanalysis of multivariate public-key cryptosystems:

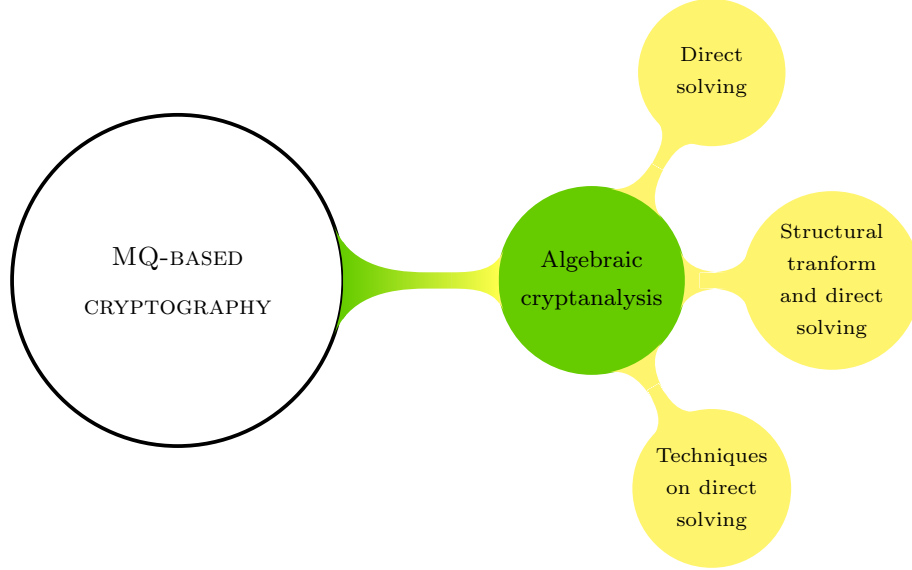
- I Direct algebraic cryptanalysis of the existing multivariate public-key cryptosystems.
- II Algebraic cryptanalysis of existing multivariate public-key cryptosystems through structural transform and direct algebraic techniques.
- III More techniques on solving multivariate quadratic polynomials, especially the technique of transforming a polynomial system over a finite field to a new polynomial system over its subfield.

1.3 Contribution

Evaluating the security of EFC using algebraic techniques (In chapter 4)

Extension field cancellation (EFC) [SDP16] is a multivariate encryption scheme and its security evaluation is through evaluating the complexity of the algebraic attack against EFC. Since the construction of EFC is similar to the construc-

Figure 1.1: Research directions on algebraic cryptanalysis on multivariate public-key cryptography



tion of HFE [Pat96] with a minus modifier (deleting a few polynomials from a public key), [SDP16] estimated its degree of regularity based on the results in [DG10, DK12, DH11, DY13] and gave security parameters for 80-bit security. However, the analysis in [SDP16] does not consider using *field equations* as all variables in EFC are over \mathbb{F}_2 , and *hybrid approach* can be very effective for polynomials over \mathbb{F}_2 as specifying every variables only costs a complexity of 2. Based on these observations, we carried out experiments on parameters of EFC with hybrid approach and concluded the proposed parameters were not secure. Finally, we estimated secure parameters for EFC. This result was published on *IEICE transactions on Fundamentals of Electronics, Communication and Computer Sciences* under the title “The secure parameters and efficient decryption algorithm for multivariate public key cryptosystem EFC” [WIDT19].

On the algebraic aspects of solving the minrank problem (In chapter 5)

The minrank problem, which searches for a linear combination of a set of matrices over a finite field with a target rank, is also related to the security of some multivariate public-key cryptosystems. Interestingly, by considering different mathe-

mathematical model, the minrank problem can be solved by solving different polynomial systems derived from different models. Minors modeling [BFP13] and Kipnis-Shamir method [KS99] are two different methods for solving the minrank problem. Complexity analysis of minors modeling is available at [CG19, FDS13, FDS10], and complexity of Kipnis-Shamir method is available at [VBC⁺19, FDS11]. The polynomial systems derived from the Kipnis-Shamir method can be divided into several subsystems, in [VBC⁺19], the degree of regularity of the full polynomial system is given by analyzing its non-trivial syzygies, which still leaves the question, that which subsystem is the best in the sense of complexity, open. For minors modeling, a large number of minors of a matrix with linear polynomial entries have to be computed. And for Kipnis-Shamir method, many extra variables have to be introduced to form a polynomial system. Combining these two methods, I proposed a mixed method, which uses a subsystem from a Kipnis-Shamir method and a subsystem from minors modeling. Moreover, I revisited the minrank attack on Rainbow [DS05]. This result was published on the proceedings of the conference *The 21st World Conference on Information Security Application* under the title “Revisiting the Minrank Attack on Multivariate Cryptography” [WINT20].

Algebraic cryptanalysis of multivariate encryption scheme PERN (In chapter 6)

The security of multivariate public-key cryptography is based on the hard of solving the MQ problem. In [Yas18], a new variant of the MQ problem called the constrained MQ problem is used on constructing multivariate encryption scheme, which is called *pq-method*. In the key construction of *pq-method*, an easy-to-invert polynomial is still used. To erase this map, [YWT20] proposes a new multivariate encryption scheme called PERN, which uses two random polynomial maps and an affine map in its private key. Different from conventional multivariate encryption schemes, its decryption is done through numerical techniques. To estimate secure parameters, we conduct algebraic cryptanalysis on PERN and deduct its degree of regularity through its Hilbert series in this thesis. This result is published in the section 5.5 of the paper at conference *Post-Quantum Cryptography – PQCrypto 2020* titled “Multivariate Encryption Schemes Based on Polynomial Equations over Real Numbers” [YWT20].

On the Weil descent attack against the MQ problem (In chapter 7)

The Weil descent attack transforms a polynomial system over a finite field to a new polynomial system over its subfield, it was first used to break the discrete logarithm problem on algebraic curve over composite fields [CP12, FPPR12, FPPR11, FR94]. It can also be applied to the MQ problem. For example, when the Weil descent is applied to a polynomial system of n polynomials in n variables over \mathbb{F}_{2^q} , we obtain a new polynomial system of nq polynomials in nq variables over \mathbb{F}_2 . Additionally, we have trivial relations on new variables, which are $x_i^2 - x_i = 0$ for $i = 1, \dots, nq$. Since the new polynomial system is not random, its degree of regularity is intractable, we analyzed its first fall degree in this thesis. By analyzing the non-trivial syzygies of the new polynomial system, we gave a concrete formula for estimating its first fall degree. Experiments on small parameters confirmed the correctness of our formula, but the degree of regularity appeared to be larger than the first fall degree in some cases. Our estimation on the first fall degree can be used as a lower bound on the degree of regularity. This result is still not published yet.

1.4 Organization

In chapter 2 This chapter explains definition, construction and classification of multivariate public-key cryptography. In addition, it includes two of the most multivariate signature schemes Rainbow and HFEv-, their construction and security analysis are introduced.

In chapter 3 This chapter explains some basics in commutative algebra and Gröbner basis. Algorithms for computing Gröbner basis and their complexity are also introduced.

In chapter 4 This chapter gives algebraic cryptanalysis of EFC. This chapter starts with explaining the construction of EFC. It then recalls the efficient decryption algorithm proposed in [WIDT18]. Then it gives the main result of this chapter, which is applying hybrid attack on EFC and breaking the proposed EFC parameters.

In chapter 5 This chapter discusses minrank problem derived in multivariate public-key

cryptosystems. First it recaps existing methods for solving the minrank problem. Then it presents a proposed mixed method for solving the method. Finally, it revisits the minrank attack on Rainbow [DS05].

In chapter 6 This chapter gives algebraic cryptanalysis of a multivariate encryption scheme PERN, which is short for polynomial equations over real numbers. It first recalls the construction of PERN, then it gives an analysis on the complexity of the algebraic attack against PERN.

In chapter 7 This chapter mainly considers the Weil descent attack on the MQ problem, which transforms a polynomial system over a finite field into a new polynomial system over its subfield. It starts with descriptions on applying the Weil descent attack on the MQ problem, then gives a detailed analysis of its first fall degree through analyzing its non-trivial syzygies.

In chapter 8 This chapter concludes the thesis and gives future work.

Chapter 2

Multivariate public-key cryptography

A multivariate public-key cryptosystem (MPKC) refers to a public-key cryptosystem whose public key consists of a set of multivariate polynomials (often quadratic) over a finite field, whose security, hence, relies on solving a set of multivariate quadratic polynomials over a finite field. This is commonly recognized as the multivariate quadratic problem (MQ problem). The goal in this field is to create a MPKC that is sufficiently secure and with as good efficiency performances as possible.

Common MPKCs do not have thorough security proof, which leaves us to evaluate their security with various applicable attacks, and the security of a MPKC is deducted from the complexity of the most powerful attack against it. The identification schemes proposed in [SSH11] are the only exception in the MPKC kingdom that has a security proof. Following this clan of research, a multivariate signature scheme, MQDSS [CHR⁺16, CHR⁺17, CHR⁺19] was proposed. However, some new cryptanalysis [KZ20] proved it as a failed attempt.

Since the first MPKC, MI [MI88] was proposed, there have been many attempts on building secure multivariate encryption and signature scheme, but multivariate signature schemes have always been more successful compared to multivariate encryption schemes. In the signature category, UOV [KPG99], SFLASH [PCG01a], Quartz [PCG01b], Rainbow [DS05], PFLASH [DDY⁺08], Gui [PCY⁺15], HMFev [PCDY17], HiMQ-3 [SPK17, SPKK17], GeMSS [CFMR⁺17, CFMR⁺19, CFMR⁺20] are proposed. Among them, UOV, Rainbow and GeMSS withstood various attacks and still remain secure. Especially, Rainbow and GeMSS are both submitted to the NIST PQC standardization project, and Rainbow proceeds to the 3rd round finalists and GeMSS is selected as an alternate candidate. As for the encryption schemes, MI, HFE [Pat96],

Square [CBD⁺09], ABC [TDTD13], ZHFE [PBD14], SRP [YS16], EFC [SDP16], HFERP [IPST⁺18], EFLASH [CST18], PERN [YWT20] are proposed. Besides ABC and recently proposed HFERP, EFLASH and PERN, the rest of multivariate encryption are broken.

Since the public key polynomials of a MPKC are usually generated from some structured private keys, when evaluating its security, two types of attacks are considered, direct attack and structural attacks. Direct attack solves the equations system yielded from a public key and a ciphertext for the encryption case or a public key and a message for the signature case using algebraic techniques. In this thesis, only Gröbner basis techniques and the XL algorithm are considered. As for the structural attack, there are several conventional structural attacks often being considered, which are UOV-related attacks [KS98, KPG99], linearization attack [KS99], rank attacks [CSV93b, VBC⁺19, CG19, FLdVP08, Wol04], differential attacks [DFSS07, FGS05, MPST14].

MPKCs, compared to other public-key cryptosystems, are very easy to implement on low cost devices since finite fields used are relatively small and arithmetic operations are simple. Moreover, very small signatures can be generated using multivariate signature schemes, which benefits multivariate signature schemes from being suitable for practical use. Conversely, to achieve the required security, a large number of multivariate polynomials have to be used, which results in large private and public keys. Finally, thorough security proofs are still missing.

Summarizing all the research work in this field, research topics are often orbiting around security analysis, key size reduction, new cryptographic schemes proposals and security proofs.

Throughout this chapter, \mathbb{F}_q denotes a finite field of q elements, $\mathbf{x} = (x_1, \dots, x_n)$ are n variables over \mathbb{F}_q and $R := \mathbb{F}_q[\mathbf{x}]$ is the polynomial ring in variables \mathbf{x} and with coefficients in \mathbb{F}_q .

In this chapter, we will also be using *matrix representation* of multivariate quadratic polynomials, which is defined as below.

Definition 2.0.1 (Matrix representation). Given a multivariate quadratic polynomial $f \in \mathbb{F}_q[\mathbf{x}]$, its matrix representation is given as follows:

$$f = (\mathbf{x} \ 1) \cdot M \cdot \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix},$$

where $M \in \mathbb{F}_q^{(n+1) \times (n+1)}$ is an upper triangular matrix, and

- for $1 \leq i \leq j \leq n$, (i, j) -th entry stores the coefficient of the term with $x_i x_j$,
- for $1 \leq i \leq n$, $(i, n+1)$ -th entry stores the coefficient of the term with x_i ,
- $(n+1, n+1)$ -th entry stores the constant of f .

Similarly, if we are considering a matrix representation for a homogeneous quadratic polynomial, we obtain a $n \times n$ matrix.

For rank type attacks against multivariate cryptosystems, symmetric matrices associated to the quadratic form of a multivariate quadratic polynomial need to be considered, which is defined as follows.

Definition 2.0.2 (Symmetric matrix associated to a quadratic polynomial). Let $f \in \mathbb{F}_q[\mathbf{x}]$ be a multivariate quadratic polynomial, \tilde{f} be the quadratic component of f and M be the associated matrix representation of \tilde{f} , then the symmetric matrix associated to f is

$$\begin{cases} \frac{1}{2} \cdot (M + M^\top), & 2 \nmid q \\ M + M^\top. & 2 \mid q \end{cases}$$

2.1 Multivariate quadratic problem

Definition 2.1.1 (MQ problem). Let $F = (f_1, \dots, f_m) \in R^m$ be a set of m polynomials in n variables. Find $\mathbf{z} \in \mathbb{F}_q^n$ such that

$$f_1(\mathbf{z}) = \dots = f_m(\mathbf{z}) = 0.$$

This problem is proven to be NP-complete [RGSJ79], and there hasn't been found any efficient quantum algorithms for solving it, which is why multivariate public-key cryptography is considered quantum resistant. To practically understand the hardness of this problem, a contest called Fukuoka MQ Challenge (<https://www.mqchallenge.org>) [YDH⁺15b] is being held. Problems of solving 6 different types of polynomial systems are published for challenges (Table 2.1), which are created based on real public key polynomial systems from MPKC.

Commonly used methods for solving multivariate polynomial systems over finite fields include, Gröbner basis technique and XL algorithm. Details about Gröbner basis techniques and XL algorithm are given in Chapter 3.

Private key

Table 2.1: 6 types in the MQ challenge

Type	Scheme	n, m	Finite Field
I	Encryption	$m = 2n$	\mathbb{F}_2
II	Encryption	$m = 2n$	\mathbb{F}_{2^8}
III	Encryption	$m = 2n$	\mathbb{F}_{31}
IV	Signature	$n \approx 1.5m$	\mathbb{F}_2
V	Signature	$n \approx 1.5m$	\mathbb{F}_{2^8}
VI	Signature	$n \approx 1.5m$	\mathbb{F}_{31}

- An easy-to-invert quadratic polynomial map (often called central map) $F = (f_1, \dots, f_m)$ that consists of a set of easily solvable multivariate quadratic polynomials:

$$F : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$$

$$\mathbf{v} \mapsto (f_1(\mathbf{v}), \dots, f_m(\mathbf{v})).$$

“Easy-to-invert” in here means the polynomials in this polynomial map can be efficiently solved and time cost is very low that can be used in building cryptographic systems with efficient decryption or signature generation.

- An invertible affine map S :

$$S : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$$

$$\mathbf{x} \mapsto M_S \mathbf{x} + \mathbf{c}_S,$$

where $M_S \in \mathbb{F}_q^{n \times n}$ is an invertible $n \times n$ matrix, $\mathbf{c}_S \in \mathbb{F}_q^n$ is a vector.

- An invertible affine map T :

$$T : \mathbb{F}_q^m \rightarrow \mathbb{F}_q^m$$

$$\mathbf{x} \mapsto M_T \mathbf{x} + \mathbf{c}_T,$$

where $M_T \in \mathbb{F}_q^{m \times m}$ is an invertible matrix, $\mathbf{c}_T \in \mathbb{F}_q^m$ is a vector.

Public key

- A composition of T , F and S given by $P(\mathbf{x}) = T \circ F \circ S(\mathbf{x}) = T(F(S(\mathbf{x})))$, which is a quadratic polynomial map that consists of m polynomials $(p_1, \dots, p_m) \in R^m$:

$$P : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$$

$$\mathbf{v} \mapsto (p_1(\mathbf{v}), \dots, p_m(\mathbf{v})).$$

Equivalent keys

- For many multivariate encryption and signature schemes, there exist equivalent keys [WP05], which relates to the security of a MPKC. They are defined as follows.

Definition 2.1.2 (Equivalent keys). Given a key pair for a MPKC as

$$(\{T, F, S\}, P),$$

the set of maps $\{T', F', S'\}$ is called an *equivalent secret key* if both of the following conditions hold:

- $T \circ F \circ S = T' \circ F' \circ S' = P$,
- F' and F have the same structure.

Given a key pair for a MPKC as $(\{T, F, S\}, P)$, let $\Sigma : \mathbb{F}_q^m \rightarrow \mathbb{F}_q^m$ and $\Omega : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ be two invertible affine transformations, then

$$P = T \circ F \circ S = \underbrace{T \circ \Sigma^{-1}}_{T'} \circ \underbrace{\Sigma \circ F \circ \Omega}_{F'} \circ \underbrace{\Omega^{-1} \circ S}_{S'}$$

holds. If $\Sigma \circ F \circ \Omega$ can be used for decryption or signature generation using algorithms for F , then $\{T', F', S'\}$ is an equivalent private key of $(\{T, F, S\})$.

Encryption scheme Encryption algorithm and decryption algorithm of a multivariate encryption scheme are shown respectively in Algorithm 2.1 and 2.2. A workflow is shown in Figure 2.1.

Algorithm 2.1: Encryption algorithm for a multivariate encryption scheme

Input : A plaintext $\mathbf{m} \in \mathbb{F}_q^n$ and a public key $P = (p_1, \dots, p_m)$

Output: A encrypted ciphertext $\mathbf{c} \in \mathbb{F}_q^m$ to the plaintext \mathbf{m}

1 $\mathbf{c} \leftarrow (p_1(\mathbf{m}), \dots, p_m(\mathbf{m}))$

2 **Return** \mathbf{c} .

Algorithm 2.2: Decryption algorithm for a multivariate encryption scheme

Input : A ciphertext $\mathbf{c} \in \mathbb{F}_q^m$ and a private key $\{T, F, S\}$
Output: A decrypted plaintext $\mathbf{m} \in \mathbb{F}_q^n$ to the ciphertext \mathbf{c}

 1 $\mathbf{d} \leftarrow T^{-1}(\mathbf{c})$

 2 $\mathbf{z} \leftarrow F^{-1}(\mathbf{d})$

 3 $\mathbf{m} \leftarrow S^{-1}(\mathbf{z})$

 4 **Return** \mathbf{m} .

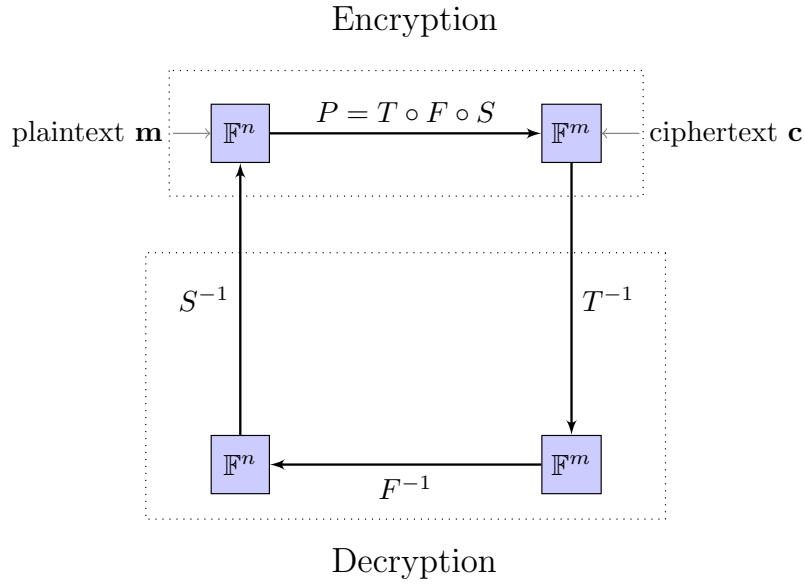


Figure 2.1: Multivariate Encryption

Signature scheme Signature generation algorithm and verification algorithm of a multivariate signature scheme are shown respectively in Algorithm 2.3 and 2.4. A workflow is shown in Figure 2.2.

2.2 The Rainbow signature scheme

In this section, we take a look at one of the most famous multivariate signature scheme called Rainbow signature scheme [DS05, DCP⁺17b, DCP⁺19, DCP⁺20]. We recall its construction and security.

Algorithm 2.3: Signature generation algorithm for a multivariate signature scheme

Input : A message to be signed $\mathbf{m} \in \mathbb{F}_q^m$ and a private key $\{T, F, S\}$

Output: A signature $\mathbf{s} \in \mathbb{F}_q^n$ to the message \mathbf{m}

1 $\mathbf{d} \leftarrow T^{-1}(\mathbf{m})$

2 $\mathbf{z} \leftarrow F^{-1}(\mathbf{d})$

3 $\mathbf{s} \leftarrow S^{-1}(\mathbf{z})$

4 **Return** \mathbf{s}

Algorithm 2.4: Signature verification algorithm for a multivariate signature scheme

Input : A message $\mathbf{m} \in \mathbb{F}_q^m$, a signature $\mathbf{s} \in \mathbb{F}_q^n$ and a public key

$$P = (p_1, \dots, p_m) \in R^m$$

Output: Boolean value for the validity of the signature

1 **if** $\mathbf{m} = P(\mathbf{s})$ **then**

2 **Return** True

3 **else**

4 **Return** False

2.2.1 Construction

Parameters and notations

· $v, o_1, o_2 \in \mathbb{N}, n := v + o_1 + o_2, m := o_1 + o_2$.

Private key

· An invertible affine map S :

$$\begin{aligned} S : \mathbb{F}_q^n &\rightarrow \mathbb{F}_q^n \\ \mathbf{x} &\mapsto M_S \mathbf{x} + \mathbf{c}_S, \end{aligned}$$

where $M_S \in \mathbb{F}_q^{n \times n}$ is an invertible $n \times n$ matrix, $\mathbf{c}_S \in \mathbb{F}_q^n$ is a vector.

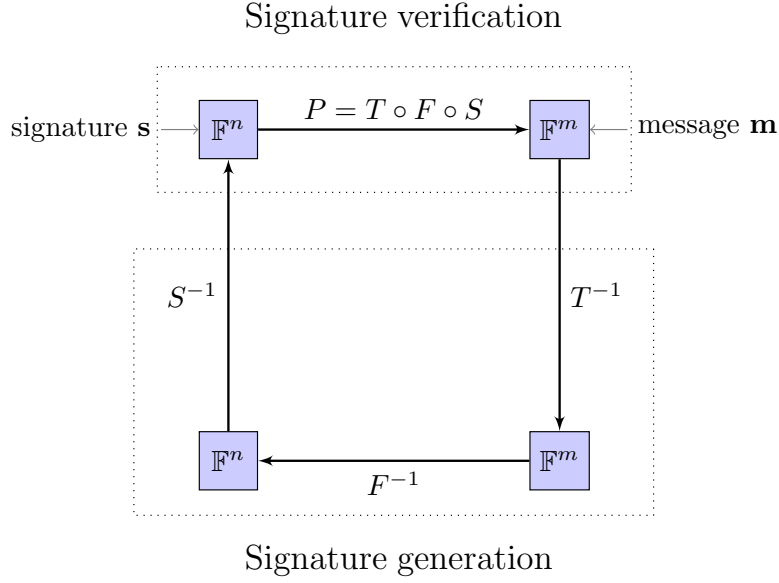


Figure 2.2: Multivariate signature

- An invertible affine map T :

$$T : \mathbb{F}_q^m \rightarrow \mathbb{F}_q^m$$

$$\mathbf{x} \mapsto M_T \mathbf{x} + \mathbf{c}_T,$$

where $M_T \in \mathbb{F}_q^{m \times m}$ is an invertible $m \times m$ matrix, $\mathbf{c}_T \in \mathbb{F}_q^m$ is a vector.

- An quadratic polynomial map $F = (f_1, \dots, f_m) :$

$$F : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$$

$$\mathbf{v} \mapsto (f_1(\mathbf{v}), \dots, f_m(\mathbf{v})),$$

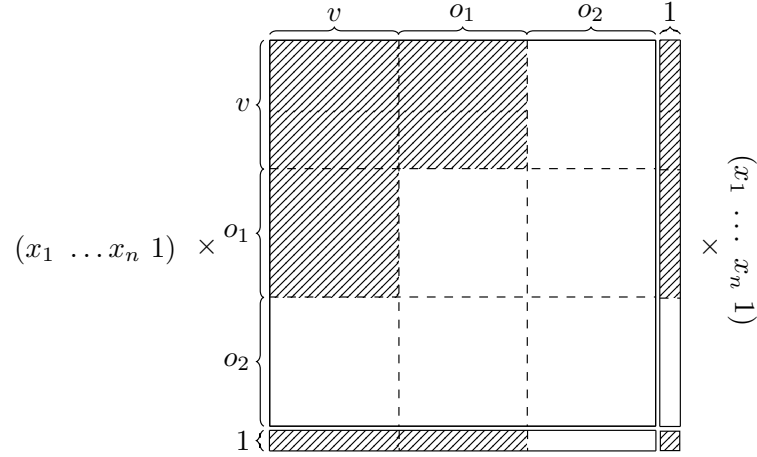
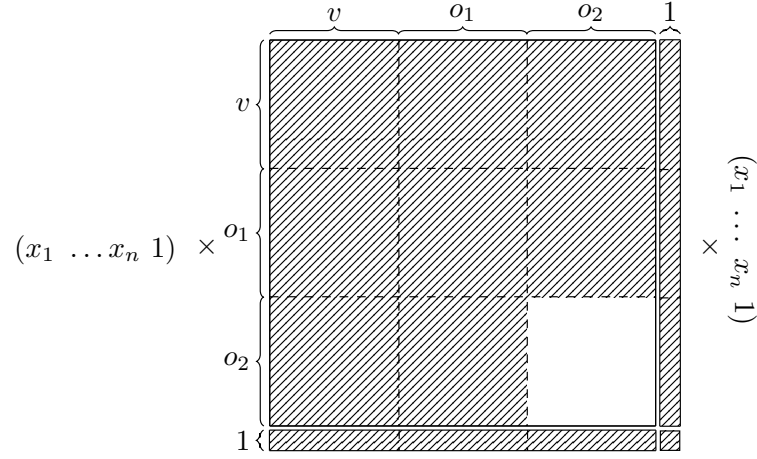
where f_1, \dots, f_{o_1} are in the form of Figure 2.3 and f_{o_1+1}, \dots, f_m are in the form of Figure 2.4. White blocks in Figure 2.3 and Figure 2.4 represent zero matrices and blocks with lines represent random matrices.

Public key

- A quadratic polynomial map from a composition of T, F, S . Namely, $P = (p_1, \dots, p_m) = T \circ F \circ S :$

$$P : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$$

$$\mathbf{v} \mapsto (p_1(\mathbf{v}), \dots, p_m(\mathbf{v})).$$


 Figure 2.3: Polynomials f_1, \dots, f_{o_1} in the private key F of Rainbow

 Figure 2.4: Polynomials f_{o_1+1}, \dots, f_m in the private key F of Rainbow

Signature generation

- Given a message $\mathbf{m} \in \mathbb{F}_q^m$ and a private key $\{T, F, S\}$, to generate a signature, we run Algorithm 2.5.

Signature verification

- Given a message $\mathbf{m} \in \mathbb{F}_q^m$, a signature $\mathbf{s} \in \mathbb{F}_q^n$ and a public key $P = (p_1, \dots, p_m)$, to verify the validity of the signature \mathbf{s} , we execute Algorithm 2.6

Algorithm 2.5: Signature generation algorithm for Rainbow

Input : A message to be signed $\mathbf{m} \in \mathbb{F}_q^m$ and a private key $\{T, F, S\}$ **Output:** A signature $\mathbf{s} \in \mathbb{F}_q^n$ to the message \mathbf{m}

```

1  $\mathbf{d} = (d_1, \dots, d_m) \leftarrow T^{-1}(\mathbf{m})$ 
2 for  $\mathbf{v} \in \mathbb{F}$  do
3   Let

$$S_1 := \begin{cases} f_1(\mathbf{v}, x_{v+1}, \dots, x_{v+o_1}) = d_1, \\ \vdots \\ f_{o_1}(\mathbf{v}, x_{v+1}, \dots, x_{v+o_1}) = d_{o_1}. \end{cases}$$

4   if Solution of  $S_1$  is  $\emptyset$  then
5     continue
6   else
7      $\mathbf{z} \leftarrow \text{Solution of } S_1$ 
8     Let

$$S_2 := \begin{cases} f_{o_1+1}(\mathbf{v}, \mathbf{z}, x_{v+o_1+1}, \dots, x_{v+o_1+o_2}) = d_{o_1+1}, \\ \vdots \\ f_m(\mathbf{v}, \mathbf{z}, x_{v+o_1+1}, \dots, x_{v+o_1+o_2}) = d_m. \end{cases}$$

9     if Solution of  $S_2$  is  $\emptyset$  then
10       continue
10    else
11       $\mathbf{b} \leftarrow \text{Solution of } S_2$ 
12       $\mathbf{w} \leftarrow (\mathbf{v}, \mathbf{z}, \mathbf{b}) \in \mathbb{F}_q^n$ 
13  $\mathbf{s} \leftarrow S^{-1}(\mathbf{w})$  Return  $\mathbf{s}$ 

```

2.2.2 Cryptanalysis

The security of Rainbow is evaluated via various different attacks such as direct attack [Buc65], minrank attack [BG06], highrank attack [Wol04], rectangular minrank attack [War20], rainbow band separation attack [DYC⁺08], UOV attack [KS98, KPG99] and

Algorithm 2.6: Signature verification algorithm for Rainbow

Input : A message $\mathbf{m} \in \mathbb{F}_q^m$, a signature $\mathbf{s} \in \mathbb{F}_q^n$ **Output:** Boolean value for the validity of the signature

```

1 if  $\mathbf{m} = P(\mathbf{s})$  then
2   | Return True
3 else
4   | Return False

```

UOV reconciliation attack [DYC⁺08].

Direct attack (with hybrid approach) Generic attack that works on all MPKCs, it requires one to solve a system of algebraic equations directly. In Rainbow, given a message $\mathbf{m} \in \mathbb{F}_q^m$ and a public key $P = (p_1, \dots, p_m) \in R^m$, the solutions to the system

$$p_1 = m_1, \dots, p_m = m_m \quad (2.1)$$

are valid signatures. This is a polynomial system of m polynomials in n variables (underdetermined), which expects to have multiple solutions. By specifying extra $n - m$ variables (direct attack) or $> n - m$ variables (direct attack with hybrid approach), the system becomes determined or overdetermined, and expects to have only one solution or no solution. The complexity of this attack is equal to solving an m polynomials in $\leq m$ variables polynomial system resulted from specifying extra $n - m$ or more variables. According to [DCP⁺17b, DCP⁺19], this system behaves very similar to a random system (semi-regular, see Definition 3.2.15), which allows us to estimate its degree of regularity (d_{reg} , see Definition 3.3.15) using

$$\frac{(1 - t^2)^m}{(1 - t)^{m-k}},$$

where k indicates $n - m + k$ variables are specified in Equation 2.1 and d_{reg} is the degree of its first non-positive term. Using an XL Wiedemann approach algorithm for solving such polynomial systems expects to have a complexity of

$$O\left(q^k \cdot 3 \cdot \binom{m - k + d_{reg}}{d_{reg}}^2 \cdot \binom{m - k}{2}\right).$$

Highrank attack Highrank identifies the variables appearing the least number of times in the central map, which, in two layered Rainbow, are x_{v+o_1+1}, \dots, x_n . Identifying those variables means recovering the subspace, $S^{-1}(\mathcal{O}_2)$, where

$$\mathcal{O}_2 := \{(x_1, \dots, x_n) \mid x_1 = 0, \dots, x_{v+o_1} = 0\}.$$

Let $(F_1, \dots, F_m) \in \mathbb{F}_q^{n \times n}$ and $(P_1, \dots, P_m) \in \mathbb{F}_q^{n \times n}$ be symmetric matrices associated to the quadratic forms of map F and P . Then we have

$$\begin{aligned} (M_S F_1 M_S^\top, \dots, M_S F_m M_S^\top) M_T &= (P_1, \dots, P_m), \\ (M_S F_1 M_S^\top, \dots, M_S F_{o_1} M_S^\top, M_S F_{o_1+1} M_S^\top, \dots, M_S F_m M_S^\top) &= (P_1, \dots, P_m) M_T^{-1}. \end{aligned}$$

Since F_1, \dots, F_{o_1} are in the form shown in Figure 2.3, given an element $\mathbf{s} \in S^{-1}(\mathcal{O}_2)$, the following relation holds:

$$\mathbf{s} \cdot M_S F_i M_S^\top = 0, \quad (1 \leq i \leq o_1),$$

which means the subspace $S^{-1}(\mathcal{O}_2)$ can be recovered from the left kernel of $\{M_S F_1 M_S^\top, \dots, M_S F_{o_1} M_S^\top\}$, and they require a partial recovery of the map T . Let $T_1 \in \mathbb{F}_q^{m \times o_1}$, $T_2 \in \mathbb{F}_q^{m \times o_2}$ be matrices such that

$$T^{-1} = [T_1 | T_2],$$

then

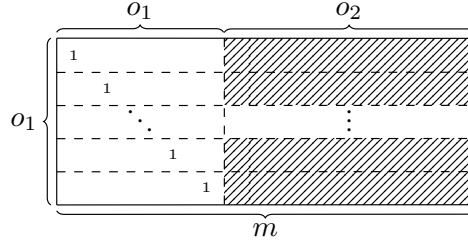
$$(P_1, \dots, P_m) T_1 = (M_S F_1 M_S^\top, \dots, M_S F_{o_1} M_S^\top), \quad (2.2)$$

$$(P_1, \dots, P_m) T_2 = (M_S F_{o_1+1} M_S^\top, \dots, M_S F_{o_1+o_2} M_S^\top). \quad (2.3)$$

Therefore, the highrank attack requires finding an echelonized basis for the subspace generated by column vectors in T_1 (See Figure 2.5) which needs a complexity of $O(q^{o_2} \cdot n^3)$.

Minrank attack Minrank attack exploits the fact that symmetric matrices associated to the quadratic forms of first layer polynomials in F have low rank. In Equation (2.2), P_1, \dots, P_m have full rank and $M_S F_1 M_S^\top, \dots, M_S F_{o_1} M_S^\top$ only have rank $v + o_1 < n$, and matrix T_1 indicates some linear combinations of P_1, \dots, P_m can result in low rank matrices. This is exactly a minrank problem instance:

$$\text{Find } (x_1, \dots, x_m) \in \mathbb{F}_q^m \text{ s.t. } \text{rank}\left(\sum_{i=1}^m x_i P_i\right) \leq v + o_1. \quad (2.4)$$

Figure 2.5: An echelonized basis of the subspace generated by column vectors in T_1

Before considering solving this instance, we need to consider the number of solutions first. The solutions are elements in the subspace generated by the column vectors of T_1 , which is a subspace of dimension o_1 , degree m and have cardinality q^{o_1} . On the other hand, if we consider the echelonized basis given in Figure 2.5, we can specify (x_1, \dots, x_m) to be $(1, 0, \dots, 0, x_{o_1+1}, \dots, x_{o_1+o_2})$, and we will only obtain one solution. Therefore, the minrank attack against Rainbow requires solving the following problem:

$$\text{Find } (1, 0, \dots, 0, x_{o_1+1}, \dots, x_m) \in \mathbb{F}_q^m \text{ s.t. } \text{rank}(P_1 + \sum_{i=o_1+1}^m x_i P_i) \leq v + o_1. \quad (2.5)$$

There are three different methods proposed for solving a minrank problem instance:

Linear algebra search Let

$$\Delta := P_1 + \sum_{i=o_1+1}^m x_i P_i,$$

this method assumes a random vector $\mathbf{v} \in \mathbb{F}_q^n$ lies in the kernel of Δ , hence

$$\mathbf{v} \cdot \Delta = \mathbf{0},$$

which gives us a linear system and solving it requires a complexity of $O(o_2^3)$. The problem left is the probability of a random vector being in the kernel of Δ . The intuitive answer would be $\frac{1}{q^{v+o_1}}$, but [BG06] showed it is $\frac{1}{q^{v+1}}$ due to the following proposition:

Proposition 2.2.1 ([BG06]). Given P_1, \dots, P_m as a public key of Rainbow,

a linear combination $\Delta = P_1 + \sum_{o_1+1}^m P_i$ of rank less than $v + o_1$ gives

$$\Delta = P_1 + \sum_{o_1+1}^m P_i = \sum_{i=1}^{o_1} b_i \cdot M_S F_i M_S^\top,$$

where $b_i \in \mathbb{F}_q$. Then the probability of a random vector lying in the kernel of Δ is $\frac{1}{q^{v+1}}$.

Proof. Let $\mathbf{v}' = M_S^\top{}^{-1} \mathbf{v} \in \mathbb{F}_q^n$ be a random vector that lies in the right kernel of Δ . Namely,

$$\Delta \cdot \mathbf{v}' = M_S \cdot \left(\sum_{i=1}^{o_1} b_i F_i \right) \cdot \mathbf{v} = 0,$$

from which we have

$$\left(\sum_{i=1}^{o_1} b_i F_i \right) \cdot \mathbf{v} = 0. \quad (2.6)$$

Suppose a vector \mathbf{v} is in the form shown in Figure 2.6, then multiplication between a matrix F_i and \mathbf{v} is shown in Figure 2.7, which is a vector with last $o_1 + o_2$ entries being 0. Therefore, Equation (2.6) means a linear combination of such vectors vanish. It has the same probability as a $v \times v$ matrix being singular, which is $\frac{1}{q}$.

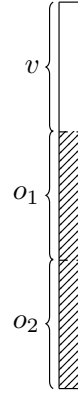
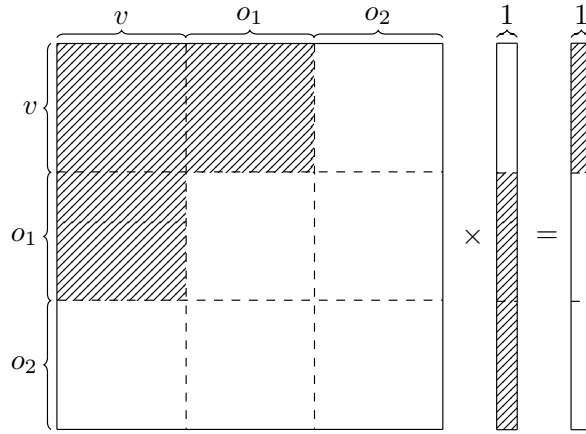
In addition, we need to consider the probability of a vector $\mathbf{v} \in \mathbb{F}_q^n$ being in the form of Figure 2.6, which is $\frac{q^{n-v}}{q^n} = \frac{1}{q^v}$.

Combining these two probabilities, we eventually have the probability of a random vector lying in the right kernel space of Δ , which is $\frac{1}{q^{v+1}}$. Therefore, the complexity of the minrank attack using linear algebra search is

$$O(q^{v+1} \cdot o_2^3).$$

□

Minors modeling [BFP13] Let $\Delta := P_1 + \sum_{i=o_1+1}^m x_i P_i$, given $\text{rank}(\Delta) \leq v + o_1$, we want to find $(1, 0, \dots, 0, x_{o_1+1}, \dots, x_m) \in \mathbb{F}_q^m$. Minor modeling utilizes the fact that $(v + o_1 + 1) \times (v + o_1 + 1)$ minors of Δ all vanish. This gives a non-linear polynomial system of $\binom{n}{v+o_1+1}^2$ equations in o_2 variables. The property of this polynomial system is related to determinantal ideals, which are intensively studied in [CG19, FDS13, FDS10]. As all $\binom{n}{v+o_1+1}^2$ polynomials are of degree $v+o_1+1$ with around $\binom{o_2+v+o_1+1}{v+o_1+1}$ monomials, this polynomial


 Figure 2.6: Form of vector \mathbf{v}

 Figure 2.7: Multiplication of F_i with \mathbf{v}

system, with high probability, has degree of regularity $d_{reg} = v + o_1 + 1$. Eventually, we obtain a complexity of the minrank attack using minor modeling as $O\left(\binom{o_2 + d_{reg}}{d_{reg}}^\omega\right)$.

Kipnis-Shamir method [KS99] The method exploits the fact that the kernel space of a rank $r = v + o_1$ size $n \times n$ matrix is of dimension $n - r$. Since Δ is of rank smaller than r , its kernel space is of dimension larger than $n - r = o_2$. Given an echelonized basis of dimension o_2 as follows:

$$K = \left(\begin{array}{cccc} \overbrace{\begin{matrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{matrix}}^{n-r} & & & \\ \hline k_1 & k_{r+1} & \cdots & k_{r(n-r-1)+1} \\ k_2 & k_{r+2} & \cdots & k_{r(n-r-1)+2} \\ \vdots & \vdots & \ddots & \vdots \\ k_r & k_{2r} & \cdots & k_{r(n-r)} \end{array} \right) \quad \begin{array}{l} \left. \vphantom{\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}} \right\} n-r \\ \left. \vphantom{\begin{pmatrix} k_1 \\ k_2 \\ \vdots \\ k_r \end{pmatrix}} \right\} r \end{array} \quad (2.7)$$

where $k_1, \dots, k_{r(n-r)}$ are variables over \mathbb{F}_q , $\Delta \cdot K = \mathbf{0}$ gives a polynomial system of $n \cdot o_2$ equations and $o_2 + o_2(v + o_1)$ variables. In addition, let $K = [\mathbf{k}_1 \ \mathbf{k}_2 \ \cdots \ \mathbf{k}_{o_2}]$, where $\mathbf{k}_i \in \mathbb{F}_q^n$, then we can obtain subsystems of $\Delta \cdot K$ from $\Delta \cdot \mathbf{k}_i = \mathbf{0}$.

The complexity of solving such polynomials system using Gröbner basis techniques or XL algorithm are discussed in [VBC⁺19, FDS11].

Support minors method [BBB⁺20, BBC⁺20] Let

$$\Delta = \begin{pmatrix} \delta_1 \\ \vdots \\ \delta_n \end{pmatrix} := \left(P_1 + \sum_{i=1}^{o_2} x_i P_{o_1+i} \right),$$

given $\text{rank}(\Delta) \leq v + o_1 = r$, there should exists r independent rows in Δ . Let C be a matrix with those independent rows and $y_1, \dots, y_{\binom{n}{r}}$ be its $r \times r$ minors. Support minors method is based on the fact that any $(r+1) \times (r+1)$ minors of

$$\begin{pmatrix} \delta_1 \\ C \end{pmatrix}, \dots, \begin{pmatrix} \delta_1 \\ C \end{pmatrix}$$

all vanish.

Every $(r+1) \times (r+1)$ minor of $\begin{pmatrix} \delta_i \\ C \end{pmatrix}$ can be obtained by using cofactor expansion on its first row and $y_1, \dots, y_{\binom{n}{r}}$, which gives a polynomial.

Therefore, the support minors method transforms a minrank problem instance from Rainbow into a $n\binom{n}{r+1}$ bilinear polynomials in $\binom{n}{r} + o_2$ variables,

$$y_1, \dots, y_{\binom{n}{r}}, x_1, \dots, x_{o_2}.$$

The complexity of this method can be obtained from the complexity of performing Wiedemann XL algorithm on the above-mentioned polynomial system. More details can be found in [BBB⁺20].

Rainbow band separation attack [DYC⁺08, Tho13] Rainbow band separation attack is a key recovery attack that recovers an equivalent key pair for Rainbow. In Definition 2.1.2, equivalent keys for a MPKC is defined. Let $\Sigma : \mathbb{F}_q^m \rightarrow \mathbb{F}_q^m$ and $\Omega : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ be two invertible affine transformations, then

$$P = T \circ F \circ S = \underbrace{T \circ \Sigma^{-1}}_{T'} \circ \underbrace{\Sigma \circ F \circ \Omega}_{F'} \circ \underbrace{\Omega^{-1} \circ S}_{S'}$$

holds. The matrices associated with Ω can only be in the form shown in Figure 2.8 since there exists relations shown in Figure 2.9 and 2.10.

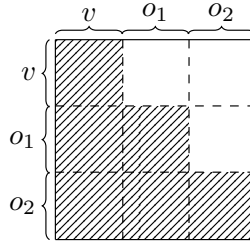


Figure 2.8: Form of the matrix M_Ω associated with Ω

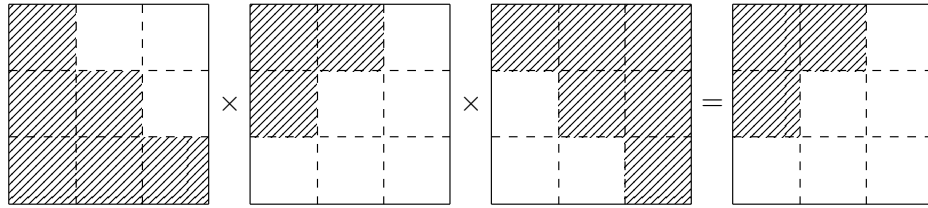
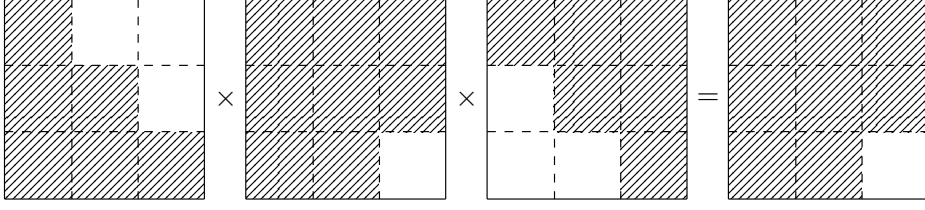
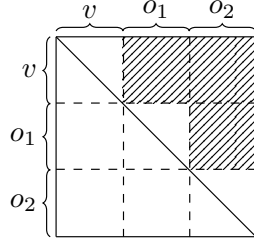


Figure 2.9: $M_\Omega F_i M_\Omega^\top$ for $i = 1, \dots, o_1$

Therefore, with high probability, there exists Ω such that the matrix associated with $S' = \Omega^{-1} \circ S$ is shown in Figure 2.11.

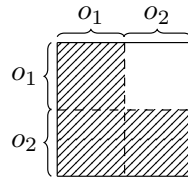
Figure 2.10: $M_{\Omega}F_iM_{\Omega}^{\top}$ for $i = o_1 + 1, \dots, m$ Figure 2.11: Form of the matrix associated with S' , diagonal line represents ones in diagonal entries

Similarly, we consider an affine map Σ whose associated matrix is in the form of Figure 2.12, and with high probability, there exists a map T' whose associated matrix is in the form of Figure 2.13.

Since $T' \circ F' \circ S' = P = T \circ F \circ S$ holds, we have

$$F' = T'^{-1} \circ P \circ S'^{-1}$$

and F' have the same construction as given in Figure 2.3 and Figure 2.4. If we regard unknown entries in $M_{S'}$ and $M_{T'}$ as variables over \mathbb{F}_q , we can obtain a polynomial system.

Figure 2.12: Form of the matrix M_{Σ} associated with Σ

Moreover, the map S' can be split into a composition of two maps S'' and Ω' ,

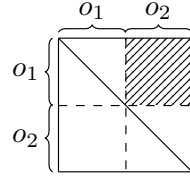


Figure 2.13: Form of the matrix $M_{T'}$ associated with $T' = T \circ \Sigma^{-1}$, the diagonal line represents ones in the diagonal entries

namely $S' = \Omega' \circ S''$ as shown in Figure 2.14, which gives

$$F' \circ S' = F' \circ \Omega' \circ S''.$$

Matrices associated to $F' \circ \Omega'$ are shown in Figure 2.15.

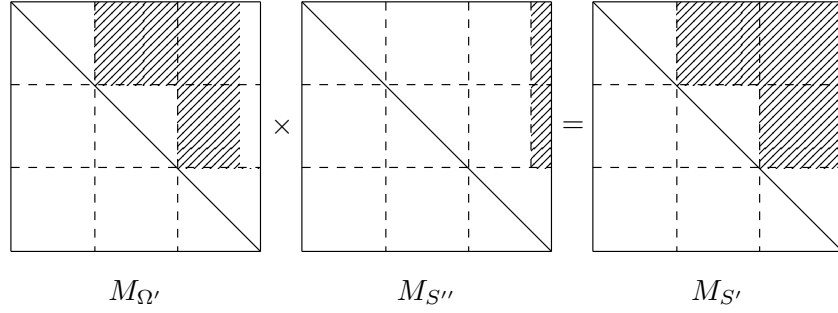


Figure 2.14: Matrix $M_{S'}$ can be split into a multiplication of two matrices $M_{\Omega'} \times M_{S''}$

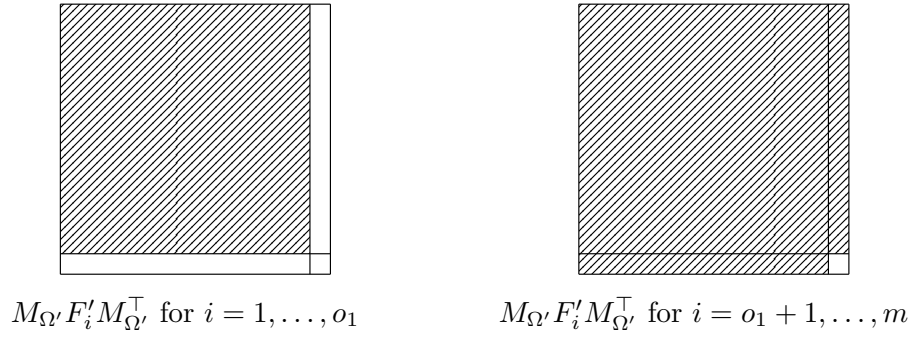


Figure 2.15: Matrix associated to $F' \circ \Omega'$

Similarly, map T' can be split into a composition of two maps, $T' = T'' \circ \Sigma'$ shown in Figure 2.16.

Eventually, we have the following relation on the public key

$$T \circ F \circ S = T \circ \Sigma^{-1} \circ \Sigma \circ F \circ \Omega \circ \Omega^{-1} \circ S = T' \circ F' \circ S' = T'' \circ \Sigma' \circ F' \circ \Omega' \circ S'' = P.$$

Matrices associated to $F' \circ \Omega'$ is already shown in Figure 2.15, and we know the form of $M_{S''}$ and $M_{T''}$, the only thing is unclear is $\Sigma' \circ F' \circ \Omega'$, which can be easily obtained. It is shown in Figure 2.17.

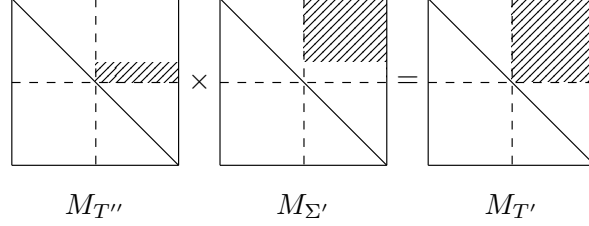


Figure 2.16: Matrix $M_{T'}$ can be split into a multiplication of two matrices $M_{T''} \times M_{\Sigma'}$

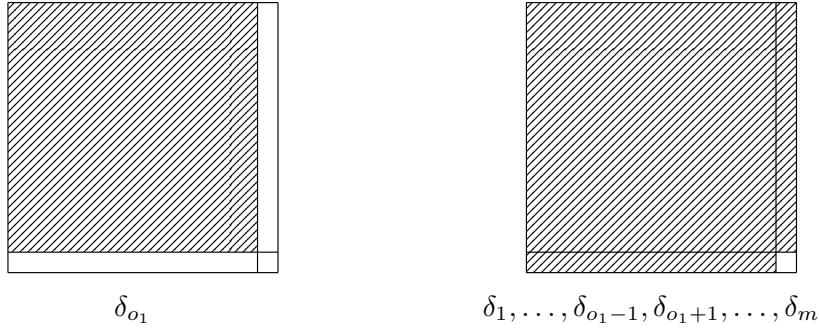


Figure 2.17: Matrix associated to $\Sigma' \circ F' \circ \Omega' = (\delta_1, \dots, \delta_m)$

To summarize, rainbow band separation attack utilizes the construction $P = T'' \circ \Sigma' \circ F' \circ \Omega' \circ S''$, where $\Sigma' \circ F' \circ \Omega'$ has a special structure presented in Figure 2.17, and S'' and T'' also have some special structure. To proceed with the attack, one regard all unknowns in $M_{S''}$ and $M_{T''}$ as variables, utilize 0 entries in the matrix representation of $\Sigma' \circ F' \circ \Omega'$ to construct a polynomial system of $n - 1$ bihomogeneous polynomials and m quadratic polynomials in n variables. Discussion on its complexity can be found in [FDS11, PST20, NIW⁺20].

UOV attack [KS98] Since Rainbow can also be regarded as UOV signature scheme with parameter $\hat{v} = v + o_1, \hat{o} = o_2$, attacks that are applicable on UOV can also

be applied to Rainbow. Let $\{\hat{F}, S\}$ be a secret key of this UOV signature scheme such that $\hat{F} \circ S = T \circ F \circ S = P$ holds.

UOV attack is a key recovery attack that exploits a notion called *invariant subspace* associated to the non-linear map \hat{F} and recovers an equivalent affine map $S' : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ such that $\hat{F}' \circ S' = \hat{F} \circ S = P$, where $\{\hat{F}', S'\}$ is an equivalent secret key. Note matrices associated with the quadratic forms of the map \hat{F} have the form shown in Figure 2.18. Let $\hat{F}_1, \dots, \hat{F}_m$ be the matrices associated to the quadratic forms of \hat{F} that are in the form of Figure 2.18.

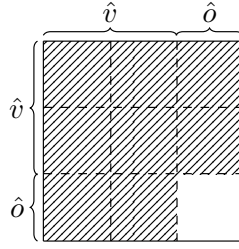


Figure 2.18: Form of the matrices associated to the quadratic forms of \hat{F}

Definition 2.2.1 (Oil space and vinegar space). Oil space is defined as

$$\mathcal{O} := \{(a_1, \dots, a_n) \in \mathbb{F}_q^n \mid a_1 = \dots = a_v = 0\},$$

and vinegar space is defined as

$$\mathcal{V} := \{(a_1, \dots, a_n) \in \mathbb{F}_q^n \mid a_{v+1} = \dots = a_n = 0\}.$$

Proposition 2.2.2. $\forall \mathbf{v}_1, \mathbf{v}_2 \in \mathcal{O}$,

$$\mathbf{v}_1 \cdot \hat{F}_i \cdot \mathbf{v}_2^\top = 0$$

for $i = 1, \dots, m$ hold. Moreover, $\forall \mathbf{u}_1, \mathbf{u}_2 \in S^{-1}(\mathcal{O})$,

$$\mathbf{u}_1 \cdot P_i \cdot \mathbf{u}_2^\top = 0$$

for $i = 1, \dots, m$ holds.

Proof. Both of these statements can be easily obtained from the structure of \hat{F} . □

The main idea of the UOV attack is to recover the subspace $S^{-1}(\mathcal{O})$, with which the public key P can be transformed into a non-linear map with the same structure as \hat{F} by Proposition 2.2.2. Moreover, when recovering the subspace $S^{-1}(\mathcal{O})$, Proposition 2.2.2 can be used to verify whether recovered subspace is correct or not.

Regarding the oil space and vinegar space, we have the following proposition.

Proposition 2.2.3. Let $E : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ be a linear map whose associated matrix is in the form of Figure 2.18. Then $E(\mathcal{O})$ is an o -dimensional proper subspace of \mathcal{V} . In addition, if E is invertible, $E^{-1}(\mathcal{V})$ is a v -dimensional subspace of \mathbb{F}_q^n and \mathcal{O} is a proper subspace of $E^{-1}(\mathcal{V})$.

Proof. These statements can be easily deducted from the definition of E , oil space and vinegar space. \square

Theorem 2.2.1 ([KPG99]). Let W_1 and W_2 be two random linear combination of P_1, \dots, P_m and W_1 is invertible. Then, with probability $q^{\hat{o}-\hat{v}}$, $W_1^{-1}W_2$ has a non-trivial invariant subspace, which is also a subspace of $S^{-1}(\mathcal{O})$.

From Theorem 2.2.1, we can recover $S^{-1}(\mathcal{O})$ by finding a common invariant subspace of linear combinations of P_1, \dots, P_m , in which vectors satisfy Proposition 2.2.2. Since invariant subspaces of a matrix A corresponds to kernel spaces of evaluation of irreducible factors of the characteristic polynomial of A at A , which cost a complexity of $O(\hat{o}^3)$ in this case. Therefore, considering the probability shown in Theorem 2.2.1, recovering a dimension \hat{o} subspace $S^{-1}(\mathcal{O})$ cost a complexity of

$$O\left(q^{(\hat{v}-\hat{o}-1)} \cdot \hat{o}^4\right) = O\left(q^{(v+o_1-o_2-1)} \cdot o_2^4\right).$$

UOV reconciliation [DYC⁺08] UOV reconciliation, analogous to the Rainbow band attack, is a key recovery attack that recovers a secret key of UOV or Rainbow using equivalent keys. Let $\{\hat{F}, S\}$ be a secret key for a UOV signature scheme with parameter $(q, \hat{v} = v + o_1, \hat{o} = o_2, m)$ transformed from Rainbow such that $\hat{F} \circ S = T \circ F \circ S = P$. Let $\Omega : \mathbb{F}^n \rightarrow \mathbb{F}_q^n$ be an invertible affine transformation that satisfies:

$$P = \hat{F} \circ S = \hat{F} \circ \Omega^{-1} \circ \Omega \circ S.$$

The matrix associated with Ω (same for Ω^{-1}) can be in the form shown in Figure 2.19 since the relation shown in Figure 2.20 holds. Moreover, with high probability, there exists Ω such that the matrix associated with $\Omega \circ S$ is in the form shown in Figure 2.21.

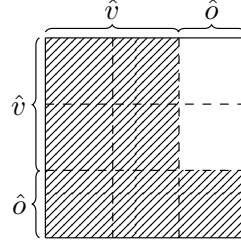


Figure 2.19: Form of the matrix M_Ω associated with Ω

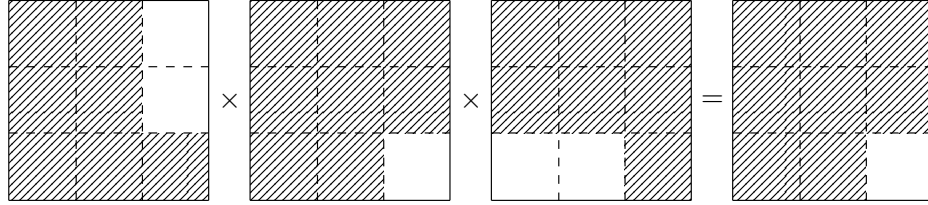


Figure 2.20: $M_\Omega \hat{F}_i M_\Omega^\top$ for $i = 1, \dots, m$

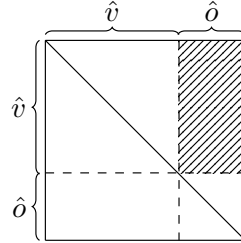


Figure 2.21: Form of the matrix associated with $\Omega \circ S$, diagonal line represents ones in diagonal entries

Let unknowns in $(\Omega \circ S)^{-1}$ be new variables, note $(\Omega \circ S)^{-1}$ is also in the form of Figure 2.21, from

$$\hat{F} \circ \Omega^{-1} = P \circ S^{-1} \circ \Omega^{-1},$$

a polynomial system of $m \frac{\hat{o}(\hat{o}+1)}{2}$ polynomials in $\hat{o}\hat{v}$ variables.

Let $S' := \Omega \circ S$, furthermore, the map S' can be expressed with a composition of \hat{o} maps, namely, $S' = S'_1 \circ S'_2 \circ \cdots \circ S'_{\hat{o}}$ just like in Figure 2.22, and we have the following equation:

$$P = \hat{F} \circ S = \hat{F}' \circ S' = \hat{F}' \circ S'_1 \circ S'_2 \circ \cdots \circ S'_{\hat{o}}, \quad (2.8)$$

which offers us many solving strategies. For example, we have

$$P \circ S'_{\hat{o}}{}^{-1} = \hat{F}' \circ S'_1 \circ \cdots \circ S'_{\hat{o}-1}, \quad (2.9)$$

where matrices associated to the quadratic form of m polynomials in $P \circ S'_{\hat{o}}{}^{-1}$ are in the form of Figure 2.23. By treating unknowns in $S'_{\hat{o}}{}^{-1}$ as variables, we can obtain a polynomial system of m polynomials in \hat{v} variables from Equation (2.9).

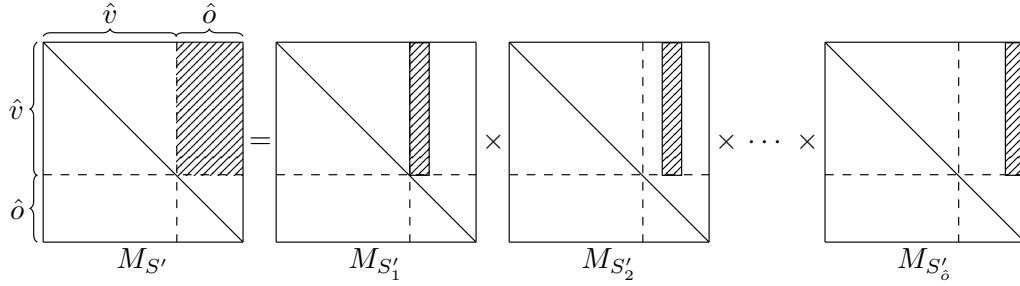


Figure 2.22: Form of the matrix associated with $\Omega \circ S$, diagonal line represents ones in diagonal entries

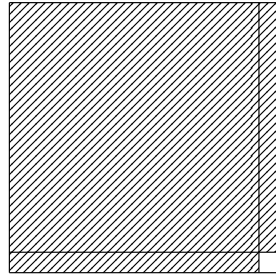


Figure 2.23: Matrices associated to the quadratic form of the m polynomials in $P \circ S'_{\hat{o}}{}^{-1}$

Similarly, we can use the following solving strategy:

$$P \circ S'_{\hat{o}}{}^{-1} \circ S'_{\hat{o}-1}{}^{-1} = \hat{F}' \circ S'_1 \circ S'_2 \circ \cdots \circ S'_{\hat{o}-2}{}^{-1}, \quad (2.10)$$

where matrices associated to the quadratic forms of m polynomials in $P \circ S'_\delta{}^{-1} \circ S'_{\delta-1}{}^{-1}$ are in the form of Figure 2.24. If we treat unknowns in $S'_\delta{}^{-1} \circ S'_{\delta-1}{}^{-1}$ as new variables, from zero entries shown in Figure 2.24, we can obtain a polynomial system of $3m$ polynomials in $2\hat{v}$ variables. Note that all polynomial system derived from Equation (2.8) can only have one solution, which means we want to choose the solving strategies that derive over-determined polynomial systems. Interestingly, since we have the relation shown in Figure 2.22, when we regard all unknown entires in $M_{S'}$ as variables $x_1, \dots, x_{\hat{v}}$, there exists a partition on those variables as $\mathbf{x} = \{x_1, \dots, x_{\hat{v}}\} = \{x_1, \dots, x_{\hat{v}}\} \cup \dots \cup \{x_{(\delta-1)\hat{v}+1}, \dots, x_{\delta\hat{v}}\}$. The complexity of solving such a multi-grading polynomial system can be derived from the multi-Hilbert series of the ideal generated by this multi-grading polynomial system.

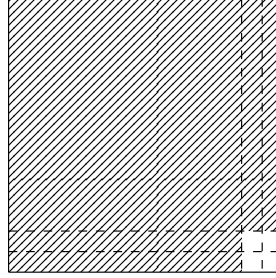


Figure 2.24: Matrices associated to the quadratic form of the m polynomials in $P \circ S'_\delta{}^{-1} \circ S'_{\delta-1}{}^{-1}$

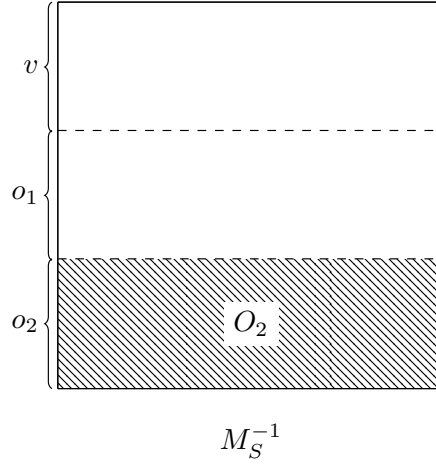
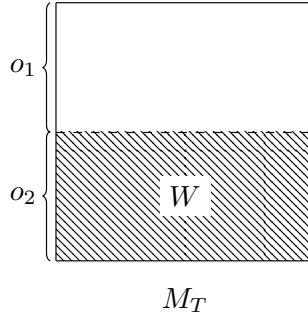
Rectangular minrank [War20] Given P_1, \dots, P_m as the symmetric matrix representation of a Rainbow public key, O_2 be a subspace of \mathbb{F}_q^n spanned by the last o_2 rows of M_S^{-1} , and W be a subspace of \mathbb{F}_q^m spanned by the last o_2 rows of M_T , shown in Figure 2.25 and Figure 2.26.

Regarding subspace O_2 and W , the following relation satisfies:

$$\{(\mathbf{x}P_1\mathbf{y}^\top, \dots, \mathbf{x}P_m\mathbf{y}^\top) \mid \mathbf{x} \in O_2, \mathbf{y} \in \mathbb{F}_q^n\} \subset W,$$

which can be easily verified from the construction of Rainbow. Therefore, for any $\mathbf{x} \in O_2$ the rank of the matrix

$$\begin{bmatrix} \mathbf{x}P_1\mathbf{e}_1^\top & \cdots & \mathbf{x}P_m\mathbf{e}_1^\top \\ \vdots & \ddots & \vdots \\ \mathbf{x}P_1\mathbf{e}_n^\top & \cdots & \mathbf{x}P_m\mathbf{e}_n^\top \end{bmatrix},$$

Figure 2.25: Subspace O_2 Figure 2.26: Subspace W

where $\mathbf{e}_i \in \mathbb{F}_q^n$ has 1 at its i -th entry and 0 elsewhere, has rank no larger than o_2 . This is a minrank problem instance, since it can be arranged into the following problem:

Problem 2.2.1 (Rectangular minrank in Rainbow). Given matrices $M_1, \dots, M_n \in \mathbb{F}_q^{n \times m}$ and a target rank o_2 , find $\mathbf{x} \in \mathbb{F}_q^n$ such that

$$\text{rank} \left(\sum_{i=1}^n x_i M_i \right) \leq o_2.$$

Since this problem has a solution space of dimension o_2 , by specifying $o_2 - 1$ variables in x_1, \dots, x_n , it expects to give only one solution. Moreover, for any $\mathbf{x} \in O_2$, $\mathbf{x} P_i \mathbf{x}^\top = \mathbf{0}$ also hold, which can be added to the rectangular problem

from Rainbow to help solving x_1, \dots, x_n using support minors method.

The complexity of this attack is given by performing Wiedemann XL algorithm on the polynomial system from the support minors method along with quadratic polynomials from $\mathbf{x}P_i\mathbf{x}^\top = \mathbf{0}$. More details can be found in [War20].

2.3 The HFEv- signature scheme

Different from Rainbow, HFEv- constructs a quadratic central map from a univariate polynomial over an extension field, Quartz [PCG01b], Gui [PCY+15, DCP+17a] and GeMSS [CFMR+17, CFMR+19] are all based on HFEv- construction. It especially offers fast verification and short signature sizes, and its public key size are medium to large compared to other cryptosystems in the post-quantum cryptography kingdom.

2.3.1 Construction

Parameters and notations

- \mathbb{F}_q : a finite field of q elements.
- \mathbb{F}_{q^n} : a degree n field extension of \mathbb{F}_q .
- $(\theta_1, \dots, \theta_n)$: a basis for $\mathbb{F}_{q^n}/\mathbb{F}_q$.
- ϕ : a linear map isomorphism between \mathbb{F}_q^n and \mathbb{F}_{q^n} given by

$$\begin{aligned} \phi : \mathbb{F}_q^n &\rightarrow \mathbb{F}_{q^n} \\ (x_1, \dots, x_n) &\mapsto \sum_{i=1}^n \theta_i x_i. \end{aligned}$$

- $D \in \mathbb{N} : q^i$ or $q^i + q^j$ for $i \neq j, i, j \leq 0$.
- $m \in \mathbb{N} : \text{the number of polynomials in a public key.}$
- $n \in \mathbb{N} : \text{the degree of a field extension of } \mathbb{F}_q$
- $v, a \in \mathbb{N} : \text{integers used in vinegar and minus modifier.}$

Key generation

Secret key

- An invertible affine map S :

$$\begin{aligned} S : \mathbb{F}_q^{n+v} &\rightarrow \mathbb{F}_q^{n+v} \\ \mathbf{x} &\mapsto M_S \mathbf{x} + \mathbf{c}_S, \end{aligned}$$

where $M_S \in \mathbb{F}_q^{(n+v) \times (n+v)}$ is an invertible $(n+v) \times (n+v)$ matrix, $\mathbf{c}_S \in \mathbb{F}_q^{(n+v)}$ is a vector.

- An invertible affine map T :

$$\begin{aligned} T : \mathbb{F}_q^n &\rightarrow \mathbb{F}_q^n \\ \mathbf{x} &\mapsto M_S \mathbf{x} + \mathbf{c}_S, \end{aligned}$$

where $M_T \in \mathbb{F}_q^{n \times n}$ is an invertible $n \times n$ matrix, $\mathbf{c}_T \in \mathbb{F}_q^n$ is a vector.

- A univariate polynomial F in variables X, y_1, \dots, y_v , where X is a variable over \mathbb{F}_{q^n} and y_1, \dots, y_v are variables over \mathbb{F}_q :

$$F = \sum_{i,j \geq 0}^{q^i + q^j \leq D} A_{i,j} X^{q^i + q^j} + \sum_{i \geq 0}^{q^i \leq D} \beta_i(y_1, \dots, y_v) X^{q^i} + \gamma(y_1, \dots, y_v), \quad (2.11)$$

where $A_{i,j} \in \mathbb{F}_{q^n}$, $\beta_i : \mathbb{F}_q^v \rightarrow \mathbb{F}_{q^n}$ are linear maps and $\gamma : \mathbb{F}_q^v \rightarrow \mathbb{F}_{q^n}$ is a quadratic map.

Public key Just like constructions of other multivariate cryptosystems, HFEv- also has public keys coming from a composition of a quadratic central map with two affine maps. We first construct this quadratic central map.

- Let I_v be the identity map on \mathbb{F}_q^v , then the map

$$\bar{F} := \phi^{-1} \circ F \circ (\phi \times I_v) : \mathbb{F}_q^{n+v} \rightarrow \mathbb{F}_n$$

is a central quadratic map for HFEv-.

- Let $\Pi : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^{n-a}$ given by

$$\begin{aligned} \Pi : \mathbb{F}_q^n &\rightarrow \mathbb{F}_q^{n-a} \\ (x_1, \dots, x_n) &\mapsto (x_1, \dots, x_{n-a}), \end{aligned}$$

be a projection.

- A public key for HFEv- is constructed from

$$P = \Pi \circ T \circ \bar{F} \circ S = \Pi \circ T \circ \phi^{-1} \circ F \circ (\phi \times I_v) \circ S,$$

which is a set of $n - a$ quadratic polynomials over \mathbb{F}_q in variables x_1, \dots, x_{n+v} .

Matrix representation To more explicitly understand the construction of HFEv-, we exploit its matrix representation.

· Let

$$M := \begin{pmatrix} \theta_1 & \theta_1^q & \theta_1^{q^2} & \cdots & \theta_1^{q^{n-1}} \\ \theta_2 & \theta_2^q & \theta_2^{q^2} & \cdots & \theta_2^{q^{n-1}} \\ \theta_3 & \theta_3^q & \theta_3^{q^2} & \cdots & \theta_3^{q^{n-1}} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \theta_n & \theta_n^q & \theta_n^{q^2} & \cdots & \theta_n^{q^{n-1}} \end{pmatrix} \quad (2.12)$$

be a matrix constructed from $(\theta_1, \dots, \theta_n)$, then the operations involved in ϕ and ϕ^{-1} can be expressed using M .

$$\begin{aligned} \phi : \mathbb{F}_q^n &\rightarrow \mathbb{F}_{q^n} \\ (x_1, \dots, x_n) &\mapsto \sum_{i=1}^n x_i M_{i,1}. \end{aligned}$$

$$\begin{aligned} \phi^{-1} : \mathbb{F}_{q^n} &\rightarrow \mathbb{F}_q^n \\ X &\mapsto \left((X, X^q, X^{q^2}, \dots, X^{q^{n-1}}) M^{-1} \right)_1. \end{aligned}$$

· $\sum_{i,j \leq 0}^{q^i + q^j \leq D} A_{i,j} X^{q^i + q^j}$ in F can be written as

$$(X, X^q, X^{q^2}, \dots, X^{q^{n-1}}) \cdot (A_{i,j}) \begin{pmatrix} X \\ X^q \\ X^{q^2} \\ \vdots \\ X^{q^{n-1}} \end{pmatrix}.$$

· Each map $\beta_i : \mathbb{F}_q^v \rightarrow \mathbb{E}$ can be written in matrix form

$$\begin{aligned} \beta_i : \mathbb{F}_q^v &\rightarrow \mathbb{F}_{q^n} \\ (y_1, \dots, y_v) &\mapsto (y_1, \dots, y_v) \cdot B_i \cdot \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_n \end{pmatrix}, \end{aligned}$$

where B_i is a $v \times n$ matrix over \mathbb{F}_q . Therefore, $\sum_{i \leq 0}^{q^i \leq D} \beta_i(y_1, \dots, y_v) X^{q^i}$

in F can be written as

$$(y_1, \dots, y_v) \cdot \left[B_1 \cdot \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} \quad B_2 \cdot \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} \quad \cdots \quad B_n \cdot \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_n \end{pmatrix} \right] \cdot \begin{pmatrix} X \\ X^q \\ X^{q^2} \\ \vdots \\ X^{q^{n-1}} \end{pmatrix}.$$

$$= (y_1, \dots, y_v) \cdot [\tilde{B}_1 \quad \tilde{B}_2 \quad \cdots \quad \tilde{B}_n] \cdot \begin{pmatrix} X \\ X^q \\ X^{q^2} \\ \vdots \\ X^{q^{n-1}} \end{pmatrix},$$

where \tilde{B}_i are $v \times 1$ matrices over \mathbb{F}_{2^n} .

- $\gamma : \mathbb{F}_q^v \rightarrow \mathbb{F}_{q^n}$ can also be expressed using matrices as

$$(y_1, \dots, y_v) \cdot (\gamma_{i,j}) \cdot \begin{pmatrix} y_1 \\ \vdots \\ y_v \end{pmatrix},$$

where $(\gamma_{i,j})$ is a $v \times v$ matrix over \mathbb{F}_{q^n} .

- Eventually, we can obtain a matrix form of F shown in Figure 2.27.

$$(X, X^q, \dots, X^{q^{n-1}}, y_1, \dots, y_v) \times \begin{pmatrix} \overbrace{\hspace{1.5cm}}^n & \overbrace{\hspace{1.5cm}}^v \\ \begin{array}{c|c} \begin{array}{c} \text{---} \end{array} & \mathbf{0} \\ \hline \begin{array}{c} \text{---} \end{array} & \begin{array}{c} \text{---} \end{array} \end{array} \\ \begin{array}{c} n \\ v \end{array} \\ \underbrace{\hspace{1.5cm}}_1 \quad \underbrace{\hspace{1.5cm}}_1 \quad \underbrace{\hspace{1.5cm}}_1 \\ M_F \end{pmatrix} \times \begin{pmatrix} X \\ X^q \\ \vdots \\ X^{q^{n-1}} \\ y_1 \\ \vdots \\ y_v \end{pmatrix}$$

Figure 2.27: Matrix representation of the univariate polynomial F used in the HFEv-, X, y_1, \dots, y_v are variables, the $(n+v) \times (n+v)$ matrix over \mathbb{F}_{q^n} in the center is denoted by M_F

Signature generation

Given a message $\mathbf{m} \in \mathbb{F}_q^{n-a}$ and a private key $\{T, F, S\}$, one can generate signature using Algorithm 2.7.

Algorithm 2.7: Signature generation algorithm for HFEv-

Input : A message to be signed $\mathbf{m} \in \mathbb{F}_q^{n-a}$ and a private key $\{T, F, S\}$

Output: A signature $\mathbf{s} \in \mathbb{F}_q^{n+v}$ to the message \mathbf{m}

```

1  $\mathbf{m}_a \leftarrow_R \mathbb{F}_q^a$ 
2  $\mathbf{w} = (w_1, \dots, w_n) \leftarrow T^{-1}(\mathbf{m}, \mathbf{m}_a)$ 
3  $W \leftarrow \phi(\mathbf{w})$ 
4 for  $\mathbf{y} = (y_1, \dots, y_v) \in \mathbb{F}$  do
5   Let  $S_1 := \{F(X, y_1, \dots, y_v) = W\}$ , solve it using Berlekamp's algorithm
   [Ber67].
6   if Solution of  $S_1$  is  $\emptyset$  then
7     continue
8   else
9      $Z \leftarrow$  a solution of  $S_1$ 
10     $\mathbf{z} \leftarrow \phi^{-1}(Z)$ 
11     $\mathbf{c} \leftarrow (\mathbf{z}, \mathbf{y}) \in \mathbb{F}_q^{n+v}$ 
12  $\mathbf{s} \leftarrow S^{-1}(\mathbf{c})$ 
13 Return  $\mathbf{s}$ 

```

Signature verification Given a signature $\mathbf{s} \in \mathbb{F}_q^{n+v}$, a message $\mathbf{m} \in \mathbb{F}_q^{n-a}$ and a public key $P = (p_1, \dots, p_{n-a})$. The validity of this signature is verified through checking whether the evaluation of the public key at \mathbf{s} is consistent with \mathbf{m} .

2.3.2 Cryptanalysis

Direct attack Direct attack solves a polynomial system derived from a public key and a message algebraically. In the case of HFEv-, this polynomial system has $n - a$ polynomials in $n + v$ variables, which is an underdetermined polynomial system, which expects to have multiple solutions. By specifying $v + a$ extra variables, this polynomial system becomes determined, and is expected to have a unique

solution. The complexity of computing a Gröbner basis for such a polynomial system is

$$O\left(\binom{n-a+d_{reg}}{d_{reg}}^\omega\right),$$

where d_{reg} is the degree of regularity (see Definition 3.3.15) of the ideal generated by this polynomial system and $2 < \omega \leq 3$ is the linear algebra constant (see Equation (3.1)).

As introduced in Section 3.2.4, the degree of regularity for regular and semi-regular polynomial systems are well understood. However, the polynomial system obtained in HFEv- is not semi-regular since its central map polynomials are of low rank (see Figure 2.27). In [DY13, DH11, DK12], theoretical upper bounds on the degree of regularity of HFE type cryptosystems are given. In the case of HFEv-, this bound is given by

$$\begin{aligned} R &:= \lfloor \log_q(D-1) + 1 \rfloor \\ d_{reg} &\leq \frac{R+v+a-1}{2} + 2 \quad R+a \text{ is odd.} \\ d_{reg} &\leq \frac{R+v+a}{2} + 2 \quad \text{otherwise} \end{aligned} \tag{2.13}$$

More recently, [Pet17] derived an lower bound on the degree of regularity of HFEv- through experiments, which is

$$d_{reg}^{ex} \geq \lceil \frac{R+a+v+7}{3} \rceil. \tag{2.14}$$

Minrank attack Matrix representation of the univariate polynomial in HFEv- is shown in Figure 2.27, we investigate more closely of the matrix M_F and the map F . Note that we have a restriction on the degree of F , which is D . This refrains matrix $(A_{i,j})$ ($0 \leq i, j \leq n$) being a full matrix. In fact, we have

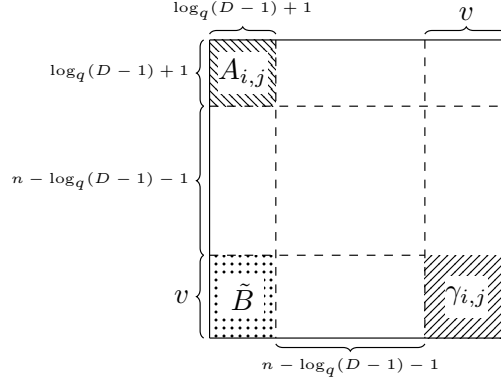
$$0 \leq i, j \leq \log_q(D-1),$$

which gives us

$$\text{rank}(A_{i,j}) \leq \log_q(D-1) + 1.$$

Similarly, restrictions on \tilde{B}_i ($1 \leq i \leq n$) can be obtained, which is $1 \leq i \leq \log_q(D) + 1$. Therefore, matrix M_F more specifically is expressed as Figure 2.28. Clearly,

$$\text{rank}(M_F) \leq \log_q(D-1) + v + 1.$$

Figure 2.28: A more precise form of M_F

Next we explore the minrank structure hidden in the public key of HFEv- using matrix representation. We already have the matrix representation of F . Let $\bar{X} := (X, X^q, \dots, X^{q^{n-1}}, y_1, \dots, y_v)$, this representation is $\bar{X} M_F \bar{X}^\top$. Let M_{I_v} be the identity matrix of $v \times v$, define

$$M_t := \begin{pmatrix} M & 0 \\ 0 & M_{I_v} \end{pmatrix}, \quad (2.15)$$

where M is defined in (2.12), then $F \circ (\phi \times I_v) \circ S$ has a matrix representation as

$$M_{FS} := M_S M_t M_F M_t^\top M_S^\top \in \mathbb{F}_{q^n}^{(n+v) \times (n+v)}. \quad (2.16)$$

After applying ϕ^{-1} on M_{FS} , we obtain n $(n+v) \times (n+v)$ matrices over \mathbb{F}_q as follows.

$$\phi^{-1}(M_{FS}) = (M_{FS}^1, \dots, M_{FS}^n) \in \left(\mathbb{F}_q^{(n+v) \times (n+v)} \right)^n,$$

which is a matrix representation of $\phi^{-1} \circ F \circ (\phi \times I_v) \circ S$. Eventually, we obtain a matrix representation for $T \circ \phi^{-1} \circ F \circ (\phi \times I_v)$ as

$$(M_P^1, \dots, M_P^n) = (M_{FS}^1, \dots, M_{FS}^n) M_T.$$

Since all matrices M_{FS}^i for $i = 1, \dots, n$ all have rank $\leq \log_q(D-1) + 1$, and all matrices M_P^1, \dots, M_P^n expects to have full rank, we know a certain linear combinations of M_P^1, \dots, M_P^n can result in a low rank matrix, and this linear combination has partial information of the private key T .

An interesting fact is that this central low rank map F is the reason that the degree of regularity of the ideal generated by public key polynomials of HFEv- is not high as semi-regular polynomial systems.

The final problem is that whether the minrank property maintains with the presence of the projection Π for deleting a polynomials.

Proposition 2.3.1 (lemma 1 [VST17]). Let $\Pi \circ T$ be a corank a linear transformation on \mathbb{F}_q^n . There exist both a nonsingular linear transformation T' and a degree q^a linear polynomial π such that $\Pi \circ T = T' \circ \phi^{-1} \circ \pi \circ \phi$.

From this proposition, we have the following relation on a public key of HFEv-: $P = \Pi \circ T \circ \phi^{-1} \circ F \circ \phi \circ S = T' \circ \phi^{-1} \circ \pi \circ F \circ \phi \circ S$, where π is a linear polynomial of degree $\leq q^a$ over \mathbb{F}_q . The composition $\pi \circ F$ is hence a quadratic polynomial of degree $\leq D \cdot q^a$, whose matrix form has rank $\leq \log_q(D - 1) + v + a + 1$. This can be used to execute the minrank attack since the linear combination of the polynomials in the public key can results in a matrix with rank less than $\log_q(D - 1) + v + a + 1$. Linear algebra search, minors modeling, Kipnis-Shamir methods and support minors method are available for solving this minrank instance similar to minrank attack on Rainbow.

Distinguishing based attack [DPPST18] In HFEv-, there are $n + v$ variables involved, which are $x_1, \dots, x_n, x_{n+1}, \dots, x_{n+v}$. Variables x_{n+1}, \dots, x_{n+v} are called vinegar variables. Define vinegar space to be

$$\mathcal{V} := (a_1, \dots, a_{n+v} \mid a_1 = \dots = a_n = 0).$$

The affine map S function as a mixer to mix vinegar variables with the rest of the variables, therefore, if the subspace $S^{-1}(\mathcal{V})$ can be recovered, vinegar variables and other variables can be separated. In this case, by adding v linearly independent linear polynomials chosen from $S^{-1}(\mathcal{V})$ to an HFEv- scheme, it turns into an HFE- scheme and subsequently attacks on HFE- can be applied [VST17]. This is the rationale behind the distinguishing based attack. More specifically, this attack observes the behavior during computing a Gröbner basis of the public key polynomial adding a few randomly chosen linear polynomials to distinguish linear polynomials in the subspace $S^{-1}(\mathcal{V})$.

2.4 Classification on MPKC

Most of the multivariate public-key cryptosystems fall into two categories, big field type schemes and small field type schemes, based on the construction of its central quadratic map.

Small field type schemes use only one finite field in all of its private keys such as UOV, Rainbow. Conversely, big field type schemes use two finite field, a base field \mathbb{F}_q and its field extension \mathbb{F}_{q^d} of some degree $d \in \mathbb{N}$. Central quadratic map of big field type schemes are constructed using a univariate polynomial over the extension field in the form of

$$\sum_{i,j \geq 0} X^{q^i + q^j},$$

because of the isomorphism between \mathbb{F}_q^d and \mathbb{F}_{q^d} , this polynomial map is a quadratic map over the base field \mathbb{F}_q , hence it can be used to construct a multivariate public-key cryptosystem. Typical cryptosystems that fall into this category are HFEv-, HFE [Pat96], PFLASH [DDY⁺08] and SFLASH [PCG01a].

Chapter 3

Algebraic techniques for solving polynomials

In this chapter, we recall some of the most basic concepts in commutative algebra and algebraic geometry that will be used in later chapters. This chapter does not contain any original work but some existing knowledge in the literature. Most of the contents come from references such as [MS05, Eis95, AM69, CLO15, BH98], where more extensive definitions and proofs can be found.

In section 3.1, we recall definitions of rings and ideals, their dimension and degree, and show some special rings and ideals such as graded rings, quotient rings and primary ideals. In section 3.2, we introduce some existing results in commutative algebra and algebraic geometry, which will be used throughout this whole thesis. Especially, the notion of Hilbert series, affine varieties, regular and semi-regular sequences will be used to explain polynomial solving. In section 3.3, we recall the definition of a Gröbner basis and some algorithms for computing it. In section 3.4, we recall some commonly used techniques and algorithms used in polynomial solving.

Through out this chapter, \mathbb{K} denotes an algebraically closed field of characteristic zero or an algebraic closure of a finite field, $\mathbf{x} = \{x_1, \dots, x_n\}$ denote variables over \mathbb{K} and $\mathbb{K}[\mathbf{x}]$ denotes the polynomial ring with coefficients in \mathbb{K} and n variables \mathbf{x} . $\mathbb{K}[\mathbf{x}]$ is a noetherian commutative ring.

3.1 Rings and Ideals

Definition 3.1.1 (Monomial, term, total degree). For the given polynomial ring with coefficients in \mathbb{K} and variables x_1, \dots, x_n , $x_1^{a_1} \cdots x_n^{a_n}$ is called a *monomial*, and (a_1, \dots, a_n) is its *exponent*. The product of a coefficient and a monomial is called a *term*, and the *total degree* (or simply *degree*) of a term or a monomial is defined by $\sum_{i=1}^n a_i$. In particular, monomials generate a polynomial algebra as a \mathbb{K} -vector space, and terms generate a polynomial algebra as an additive group.

Definition 3.1.2 (Polynomial). A *polynomial* f in x_1, \dots, x_n with coefficients in a field \mathbb{K} is a finite linear combination of monomials, namely,

$$f = \sum_i \alpha_i x_1^{a_{1,i}} \cdots x_n^{a_{n,i}}, \quad \alpha_{i,j} \in \mathbb{K},$$

the set of all polynomials in x_1, \dots, x_n with coefficients in \mathbb{K} is $\mathbb{K}[x_1, \dots, x_n]$. Its degree is defined to be the maximal degree of its monomials, denoted by $\deg(f)$.

Definition 3.1.3 (Homogeneous polynomial). A *polynomial* f in $\mathbb{K}[x_1, \dots, x_n]$ is said to be homogeneous if all of its monomials share the same degree.

3.1.1 Ideals, rings, quotient rings and graded rings

We assume the definitions of groups, rings (with unit) and fields are known, and we consider about ideals and graded rings in this section.

Definition 3.1.4 (Ideal). Given R as a commutative ring, an *ideal* $I \subset R$ is a subset of R such that

- $\forall a \in I, \forall b \in R, ab \in I$.
- $\forall a_1, a_2 \in I, a_1 + a_2 \in I$.

Given $F = \{f_1, \dots, f_m\} \subset R$, the ideal generated by F is denoted by $\langle F \rangle := \left\{ \sum_{i=1}^m f_i g_i \mid g_1, \dots, g_m \in R \right\}$.

Definition 3.1.5 (Noetherian ring). A ring R is *noetherian* if and only if its any ascending chain of ideals

$$I_1 \subseteq \cdots I_{k-1} \subseteq I_k \subseteq I_{k+1} \subseteq \cdots$$

stabilizes after a certain number n , i.e. $I_n = I_{n+1} = \dots$.

This definition is equivalent to the following proposition.

Proposition 3.1.1 ([Eis95], Sec.1.4). A ring R is noetherian if and only if all of its ideals are finitely generated, which means there exist polynomials f_1, \dots, f_m such that $\langle f_1, \dots, f_m \rangle = R$.

Proposition 3.1.2 (Hilbert's basis theorem [Eis95]). Given that the ring R is noetherian, then the polynomial ring $R[\mathbf{x}]$ is also noetherian.

\mathbb{K} is a field, the polynomial ring $\mathbb{K}[\mathbf{x}]$ is a noetherian ring, therefore every ideal $I \subset \mathbb{K}[\mathbf{x}]$ is finitely generated by a set of polynomials.

Definition 3.1.6 (Coset). Given a ring R and an ideal $I \subset R$, an equivalence relation \sim can be defined as $a \sim b$ if and only if $a - b \in I$. This relation is a congruence relation as a and b are congruent modulo I . Given $a \in R$, all elements in R that are congruent with a modulo I are $\{a + r \mid r \in I\}$, written as $[a]_I$, which is called the *coset* of a .

All of such cosets of I in R form a ring called quotient ring since for any $a, b \in R$, $[a]_I + [b]_I = [a + b]_I$ and $[a]_I \cdot [b]_I = [a \cdot b]_I$ holds.

Definition 3.1.7 (Quotient ring). Given an ideal $I \subset R$, a new ring, *quotient ring* R/I , whose elements are the cosets of I in R subject to multiplicative and additive operations.

Definition 3.1.8 (Graded ring [Lan02], Ch. XVI, Sec. 6). A *graded ring* is a ring R together with a set of subgroups $R_d, d \geq 0$ such that the following two conditions holds:

$$\begin{aligned} \cdot R &= \bigoplus_{i=0}^{\infty} R_i, \\ \cdot R_i \cdot R_j &\subset R_{i+j}, \end{aligned}$$

where the degree of each element of R_i is defined as i , and the product of an element of degree i in R_i and an element of degree j in R_j has a degree $i + j$.

A polynomial $f \in R$ of degree d can be decomposed into $f = \sum_{i=1}^d f_i$ with $f_d \neq 0$, and $f_i \in R_i$ are called the homogeneous component of f of degree i .

To introduce the concept of the dimension of an ideal, we first take a look at several special ideals.

Definition 3.1.9 (Principle ideal). An ideal $I \subset R$ is said to be *principal* if it is generated by only one element.

Definition 3.1.10 (Proper ideal). An ideal $I \subset R$ is said to be *proper* if $I \neq R$.

Definition 3.1.11 (Prime ideal). A proper ideal $I \subset R$ is said to be *prime* if for any $a, b \in R$, $ab \in I$ implies either $a \in I$ or $b \in I$.

Definition 3.1.12 (Primary ideal). A proper ideal $I \subset R$ is said to be *primary* if for any $a, b \in R$ whenever $ab \in I$, it implies either $a \in I$ or $b^n \in I$ for some $n > 0$.

Definition 3.1.13 (Radical and radical ideal). The *radical of an ideal* $I \subset R$, denoted by \sqrt{I} , is an ideal defined as

$$\sqrt{I} = \{r \in R \mid r^n \in I \text{ for some } n \in \mathbb{N}\}.$$

An ideal I is *radical* if and only if $I = \sqrt{I}$.

\sqrt{I} is prime for a *primary* ideal I . In this case, \sqrt{I} is called the associated prime of I . The notion of a primary ideal is important in commutative ring theory because every ideal of a noetherian ring has a primary decomposition. Namely, it can be written also as a finite intersection of primary ideals.

Proposition 3.1.3 (Primary decomposition Sec. 4.8 [CLO15]). A *primary decomposition* of an ideal $I \subset R$ is an expression of I as an intersection of primary ideals:

$$I = \cap_{i=1}^r Q_i.$$

It is called *minimal* or *irredundant* if the $\sqrt{Q_i}$ are all distinct and $Q_i \not\supseteq \cap_{j \neq i} Q_j$.

We also have the following Lasker-Noether decomposition theorem.

Theorem 3.1.1 (Lasker-Noether decomposition theorem Sec. 4.8 [CLO15]). Every ideal $I \subset R$ has a minimal primary decomposition.

The minimal primary decompositions of I are not uniquely determined, but the number of primary ideals involved and their associated primes are uniquely determined. For example, given (Q_1, \dots, Q_r) as a minimal primary decomposition of I , the number of primary ideals involved, r , and the prime ideals $\sqrt{Q_1}, \dots, \sqrt{Q_r}$ are uniquely determined.

Definition 3.1.14 (Krull dimension, Ch. 9 [Eis95]). The length of a prime ideal $I \subset R$

is the supremum of all integers n such that there exists a chain $I_0 \subset I_1 \subset \cdots \subset I_n = I$ of distinct prime ideals. The *Krull dimension* of R is the supremum of the lengths of chains of distinct prime ideals in R .

This definition coincides with the dimension of a vector space over a field, which is the length of the longest chain of proper subspaces.

Definition 3.1.15 (Dimension of an ideal, Ch. 9 [Eis95]). The *dimension of an ideal* $I \subset R$ is the Krull dimension of the quotient ring R/I .

For a 0-dimensional ideal, there is a very useful proposition.

Proposition 3.1.4 (Sec. 8 [AM69]). For a 0-dimensional ideal $I \subset R$, any ascending chain of ideals in R/I

$$I_1 \subseteq \cdots \subseteq I_{k-1} \subseteq I_k \subseteq I_{k+1} \subseteq \cdots$$

stabilizes after a certain number n , i.e. $I_n = I_{n+1} = \cdots$ and R/I is a finite dimensional \mathbb{K} -vector space.

Proposition 3.1.5 (Ex. 8.3 [AM69]). An ideal $I \subset R$ is of 0-dimensional if and only if R/I is a \mathbb{K} -vector space of finite dimension.

Definition 3.1.16 (Homogeneous ideal). Let R be a graded ring and $I \subset R$ be an ideal. If I is generated by homogeneous elements, we say the ideal I is *homogeneous*.

3.2 Commutative algebra and algebraic geometry

3.2.1 Affine varieties

Definition 3.2.1 (Affine variety). Given $f_1, \dots, f_m \in \mathbb{K}[x_1, \dots, x_n]$, its *affine variety* is defined by the set

$$\mathbb{V}(f_1, \dots, f_m) := \{(a_1, \dots, a_n) \in \mathbb{K}^n \mid f_i(a_1, \dots, a_n) = 0, i = 1, \dots, m\}.$$

Proposition 3.2.1. Given $F = (f_1, \dots, f_m) \in \mathbb{K}[x_1, \dots, x_n]^m$, let $I \subset \mathbb{K}[x_1, \dots, x_n]$ be the ideal generated by F . Then we have $\mathbb{V}(F) = \mathbb{V}(I)$.

Proof. Since $F \subset I$, $\mathbb{V}(I) \subset \mathbb{V}(F)$ holds. Conversely, suppose $(a_1, \dots, a_n) \in \mathbb{V}(F)$. $\forall g \in I, \exists g_1, \dots, g_m \in \mathbb{K}[x_1, \dots, x_n]$ such that $g = \sum_{i=1}^m g_i f_i$. Therefore, $g(a_1, \dots, a_n) = 0$

and $\mathbb{V}(F) \subset \mathbb{V}(I)$. □

The affine variety of an ideal represents the geometrical side of an ideal. The set of polynomials that vanish at a set of points can also define a radical ideal.

Definition 3.2.2. Given a set of points $V \subset \mathbb{K}^n$. Polynomials that vanish at V generate an ideal $\mathbb{I}(V)$ given by

$$\mathbb{I}(V) := \{f \in \mathbb{K}[x_1, \dots, x_n] \mid f(\mathbf{a}) = 0, \text{ for all } \mathbf{a} \in V\}.$$

Proposition 3.2.2 (The Strong Nullstellensatz, Thm. 6, Sec. 4.2 [CLO15]). Let $I \subset \mathbb{K}[\mathbf{x}]$ be an ideal, then

$$\mathbb{I}(\mathbb{V}(I)) = \sqrt{I}$$

holds, \sqrt{I} is the radical of I .

Definition 3.2.3 (Algebraic set). A subset $V \subset \mathbb{K}^n$ is called an algebraic set if there exists an ideal I such that $V = \mathbb{V}(I)$.

Definition 3.2.4 (Zariski topology). \emptyset and \mathbb{K}^n are the algebraic sets given by the affine varieties of $\mathbb{K}[\mathbf{x}]$ and $\langle 0 \rangle$, respectively. For any two ideals I and J , we have $\mathbb{V}(I) \cup \mathbb{V}(J) = \mathbb{V}(I \cap J)$, namely, the intersection of any two algebraic sets is again an algebraic set. For a finite set of ideals I_i , we have $\bigcap_{i=1}^{\infty} \mathbb{V}(I_i) = \mathbb{V}(\bigcup_{i=1}^{\infty} I_i)$. Namely, any finite union of algebraic sets is an algebraic set. The *Zariski topology* over \mathbb{K}^n is the topology where the closed sets are the affine varieties and the open sets are their complements.

3.2.2 Homogenization and dehomogenization

Definition 3.2.5 (Homogenization of a polynomial). Let $f \in \mathbb{K}[\mathbf{x}]$ be a polynomial of degree d in x_1, \dots, x_n and H be a new homogenizing variable, then the *homogenization* of f is

$$f^h(H, x_1, \dots, x_n) = H^d \cdot f\left(\frac{x_1}{H}, \dots, \frac{x_n}{H}\right).$$

Similarly, if $F = (f_1, \dots, f_m)$, its homogenization is $F^h = (f_1^h, \dots, f_m^h)$.

Definition 3.2.6 (Dehomogenization of a homogenous polynomial). Let

$$g \in \mathbb{K}[H, x_1, \dots, x_n]$$

be a polynomial of degree d in H, x_1, \dots, x_n , where H is a homogenization variable, then the *dehomogenization* of g is

$$g^d(x_1, \dots, x_n) = g(1, x_1, \dots, x_n).$$

Similarly, if $G = (g_1, \dots, g_m)$, its dehomogenization is $G^d = (g_1^d, \dots, g_m^d)$.

Proposition 3.2.3. For any $f \in \mathbb{K}[\mathbf{x}]$, $(f^h)^d = f$ and for any $g \in \mathbb{K}[H, \mathbf{x}]$, if g is not divisible by H , $(g^d)^h = g$.

Definition 3.2.7 (Homogenization and dehomogenization of an ideal). Let I be an ideal in $\mathbb{K}[\mathbf{x}]$, the *homogenization* of I , denoted by I^h , is given by

$$I^h = \langle f^h \mid f \in I \rangle.$$

Let J be an ideal in $\mathbb{K}[H, \mathbf{x}]$, where H is a homogenization variable, the *dehomogenization* of J is defined as

$$J^d = \langle f^d \mid f \in J \rangle.$$

3.2.3 Hilbert series

Definition 3.2.8 (Affine Hilbert function and affine Hilbert polynomial, Sec. 9.3, Def. 2 [CLO15]). Let I be an ideal in $R = \mathbb{K}[\mathbf{x}]$, polynomials of total degree $\leq t$ in R be

$$R_{\leq t} = \mathbb{K}[\mathbf{x}]_{\leq t},$$

then $R_{\leq t}$ is a vector space of dimension $\binom{n+t}{t}$. Let $I_{\leq t} = I \cap R_{\leq t}$ denote the set of polynomials in I of total degree $\leq t$. Note that $I_{\leq t}$ is a subspace of $R_{\leq t}$. The *affine Hilbert function* of I is the function on the nonnegative integers t defined by

$$aHF_{R/I}(t) = \dim_{\mathbb{K}} R_{\leq t} / I_{\leq t} = \dim_{\mathbb{K}} R_{\leq t} - \dim_{\mathbb{K}} I_{\leq t}.$$

It is the number of monomials not in I of total degree $\leq t$ and for a sufficiently large t , it is given by a polynomial function

$$aHF_{R/I}(t) = \sum_{i=1}^d b_i \binom{s}{d-i},$$

where $b_i \in \mathbb{Z}$ and b_0 is positive. This polynomial is called the *affine Hilbert polynomial* of I and is denoted $aHP_{R/I}(t)$.

Definition 3.2.9 (Index of regularity of an ideal, Sec. 9.3 [CLO15]). Let I be an ideal of $\mathbb{K}[\mathbf{x}]$. Its affine Hilbert polynomial $aHP_{R/I}(t)$ equals its affine Hilbert function $aHF_{R/I}(t)$ when $t > t_0$ for a sufficiently large t_0 . The smallest possible t_0 is called the affine index of regularity of I , denoted by $ai_{reg}(I)$.

For homogeneous ideals, we can define their Hilbert function, Hilbert polynomial and Hilbert series.

Definition 3.2.10 (Hilbert function, Hilbert series and Hilbert polynomial, Sec. 9.3 [CLO15]). Let I be a homogeneous ideal in $R = \mathbb{K}[\mathbf{x}]$, the set of homogeneous polynomials of total degree t along with the zero polynomial in R be $R_t^h = \mathbb{K}[\mathbf{x}]_t^h$, and the set of homogeneous polynomials in I of total degree t along with the zero polynomial be $I_t = I \cap R_t^h$. Then I_t is a vector subspace of R_t^h . The *Hilbert function* of I is defined by

$$HF_{R/I}(t) = \dim_{\mathbb{K}} R_t^h / I_t,$$

which maps an integer t onto the dimension of the \mathbb{K} -vector space R_t^h / I_t . The *Hilbert series* of I is given by the formal series

$$HS_{R/I} = \sum_{n=0}^{\infty} HF_{R/I}(n)t^n.$$

Similar to the affine case, for a sufficiently large t , there exists a polynomial $HP_{R/I} \in \mathbb{Q}[t]$ such that

$$HP_{R/I}(t) = HF_{R/I}(t),$$

and it is known as the *Hilbert polynomial* of the homogeneous ideal I .

Definition 3.2.11 (Homogeneous index of regularity of a homogeneous ideal). Let I be a homogeneous ideal of $\mathbb{K}[\mathbf{x}]$. Its Hilbert polynomial $HP_{R/I}(t)$ equals its Hilbert function $HF_{R/I}(t)$ when $t > t_0$ for a sufficiently large t_0 . The smallest possible t_0 is called the *homogeneous index of regularity* of I (or *index of regularity* of I), denoted by $i_{reg}(I)$. If I is 0-dimensional, $i_{reg}(I)$ is easy to read from the Hilbert series as it is a polynomial.

Example 3.2.1. Let's consider the graded ring $\mathbb{K}[x_1, \dots, x_n]$. Its Hilbert function $HF_{\mathbb{K}[\mathbf{x}]}(d)$ is the dimension of $\mathbb{K}[\mathbf{x}]_d$ as a \mathbb{K} -vector space, which is the number of monomials of degree d , namely

$$\binom{n+d-1}{d} = \binom{n-1+d}{n-1}.$$

Therefore, the Hilbert series of $\mathbb{K}[\mathbf{x}]$ is

$$HS_{\mathbb{K}[\mathbf{x}]}(t) = \sum_{d=0}^{\infty} \binom{n-1+d}{n-1} t^d = \frac{1}{(1-t)^n}.$$

Proposition 3.2.4 ([CLO15], Sec. 10.2, Lem. 5). Let R be a graded ring and f a homogeneous element of degree d in R with the condition of not being a zero divisor. Then we have

$$HS_{R/\langle f \rangle}(t) = (1-t^d)HS_R(t).$$

Proposition 3.2.5 ([MS05], Thm. 8.20). Let $I \subset R$ be a homogeneous ideal, then there exists a polynomial $P(t) \in \mathbb{Z}(t)$ such that

$$HS_{R/I}(t) = \frac{P(t)}{(1-t)^n}.$$

Hilbert series contains many geometric and algebraic information of an homogeneous ideal, such as its dimension and degree.

Definition 3.2.12 (Degree of an ideal). Let $I \subset R$ be an ideal, and \tilde{I} be the ideal generated by the homogeneous components of highest degrees of polynomials in I . Consider the irreducible form of the Hilbert series of \tilde{I} given by

$$HS_{R/\tilde{I}} = \frac{P(t)}{(1-t)^d},$$

where d is the Krull dimension of \tilde{I} and $P(t) \in \mathbb{Q}[t]$ is a polynomial such that $P(1) \neq 0$. The *degree of I* is defined as $\deg(I) = P(1)$. When I is a 0-dimensional ideal, its degree coincides with the linear dimension of R/I .

Proposition 3.2.6 ([BH98], Ch. 4). Given $I \subset R$ as a homogeneous ideal and let

$$HS_{R/I} = \frac{P(t)}{(1-t)^d}$$

be the irreducible form of its Hilbert series. Then we have the Krull dimension of I is d . Moreover, if $d = 0$, then $HS_{R/I}$ is a polynomial and $\deg(I) = HS_{R/I}(1)$.

The degree of a 0-dimensional homogeneous ideal counts, with multiplicity, the number of solutions of the system of polynomials that generates the ideal. For the positive-dimensional case, the degree of an ideal is not necessarily the number of solutions.

3.2.4 Regular and semi-regular sequences

Definition 3.2.13 (Regular sequence, Ch. 17 [Eis95]). Given a sequence of polynomials

$$F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}],$$

it is called a *regular sequence* if for any $i \in \{1, \dots, m\}$, f_i is not a zero-divisor in

$$\mathbb{K}[\mathbf{x}] / \langle f_1, \dots, f_{i-1} \rangle.$$

Namely, for any $a \in \mathbb{K}[\mathbf{x}] \setminus \{0\}$, $af_i \notin \langle f_1, \dots, f_{i-1} \rangle$.

Note that for regular sequences the order of the polynomials in the sequence does matter. However, when the regular sequence is a homogeneous regular sequence, any permutation of its polynomials is also a regular sequence.

Definition 3.2.14 (Syzygy). Given a sequence of polynomials $F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$ of degree d_1, \dots, d_m , the *syzygy module* of F is the submodule $Syz(F) \in \mathbb{K}[\mathbf{x}]^m$ of m -tuples

$$(s_1, \dots, s_m) \in \mathbb{K}[\mathbf{x}]^m \text{ such that } \sum_{i=1}^m s_i f_i = 0.$$

Let the degrees of s_1, \dots, s_m be d_1^s, \dots, d_m^s , then the degree of a *syzygy* $\mathbf{s} \in Syz(F)$ is defined as

$$\max_{i=1, \dots, m} \{d_i^s + d_i\}.$$

For a regular sequence, all relations between elements in this sequence come from the commutative law of multiplication in the ring. Namely, for each pair (i, j) with $1 \leq i < j \leq m$ we have syzygies $\mathbf{s}_{ij} = -f_j \mathbf{e}_i + f_i \mathbf{e}_j$, where $\mathbf{e}_i = (0, \dots, 0, 1, 0, \dots, 0) \in \mathbb{K}[\mathbf{x}]^m$ is the vector with 1 at i -th position and 0 elsewhere. Syzygies in this form are called *Koszul syzygies*, which are also often considered as *trivial* syzygies. Armed with the notion of syzygies, we have the following proposition about syzygies and a regular sequence.

Proposition 3.2.7. Given a regular polynomial sequence, its syzygies are all Koszul syzygies.

Followed from Proposition 3.2.4 and Proposition 3.2.6, we can easily obtain the following proposition.

Proposition 3.2.8 ([BFSY05]). Given a homogeneous polynomial sequence

$$F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$$

with $m \leq n$. The following three statements are equivalent:

- F is a homogeneous regular sequence.
- The Hilbert series of $\mathbb{K}[\mathbf{x}]/\langle F \rangle$ is

$$HS_{\mathbb{K}[\mathbf{x}]/\langle F \rangle} = \frac{\prod_{i=1}^m (1 - t^{\deg(f_i)})}{(1 - t)^n}.$$

- The dimension of $\langle F \rangle$ is $n - m$.

Proposition 3.2.9 (Bézout bound). Let $F = (f_1, \dots, f_m) \in \mathbb{K}[x_1, \dots, x_n]^m$ be a sequence of homogeneous polynomials of degree d_1, \dots, d_m . If F is a homogeneous regular sequence, then the degree of the ideal $\langle F \rangle$ is given by the Bézout bound:

$$\deg(\langle F \rangle) = \prod_{i=1}^m d_i.$$

Moreover, when $m = n$, the system F has at most $\prod_{i=1}^m d_i$ solutions.

When a given polynomial sequence has more polynomials than variables, it can not be regular by the definition. To extend the notion of the regular sequence, semi-regular sequences are defined.

Definition 3.2.15 (Homogeneous semi-regular sequence [BFSY05] Def. 5). Let $I \subset \mathbb{K}[\mathbf{x}]$ be a homogeneous ideal over the graded ring \mathbb{K} . An element $f \in \mathbb{K}[\mathbf{x}]_d$ is called *semi-regular* on $\mathbb{K}[\mathbf{x}]/I$ if the multiplication maps

$$(\mathbb{K}[\mathbf{x}]/I)_{a-d} \xrightarrow{f} (\mathbb{K}[\mathbf{x}]/I)_a$$

are linear maps of maximal rank for all a . Namely, if there exists $g \in (\mathbb{K}[\mathbf{x}]/I)_{a-d}$ and

$$fg \in (\mathbb{K}[\mathbf{x}]/I)_a$$

then $g \in I$ for all a . A homogeneous sequence of $f_1, \dots, f_m \in \mathbb{K}[\mathbf{x}]$ with degrees d_1, \dots, d_m is called a *homogeneous semi-regular sequence* if f_i is semi-regular on $\mathbb{K}[\mathbf{x}]/\langle f_1, \dots, f_{i-1} \rangle$ for all $i = 1, \dots, m$.

Definition 3.2.16 (Affine semi-regular sequence [BFSY05]). Let $F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$ be a polynomial system. F is called an *affine semi-regular sequence* if the system of homogeneous parts of highest degrees of F is a homogeneous semi-regular sequence.

Regarding semi-regular sequences, we have the following proposition.

Proposition 3.2.10 ([BFSY05]). Let $F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$ be a homogeneous semi-regular sequence with degrees d_1, \dots, d_m and $m \geq n$. Then the Hilbert series of $\mathbb{K}[\mathbf{x}]/\langle F \rangle$ is given by

$$HS_{\mathbb{K}[\mathbf{x}]/\langle F \rangle} = \left[\frac{\prod_{i=1}^m (1 - t^{d_i})}{(1 - t)^n} \right]_+,$$

where $[a]_+$ means the series obtained by truncating a at its first non-positive coefficient. Moreover, the ideal $\langle F \rangle$ has dimension 0 and syzygies of F of degree at most $\deg(HS_{\mathbb{K}[\mathbf{x}]/\langle F \rangle})$ are Koszul syzygies generated from $\langle f_i \mathbf{e}_j - f_j \mathbf{e}_i \rangle$.

In cryptography, polynomial systems with more polynomials than variables (also called overdetermined polynomial systems) are asked to be solved. In practice, random polynomial systems are observed to be semi-regular, but this has not been proved, and is stated by the Fröberg's conjecture [Frö85].

Conjecture 3.2.1 (Fröberg's conjecture). Let (d_1, \dots, d_m) be a set of degrees. Then there exists at least one semi-regular sequence of homogeneous polynomials with degree d_1, \dots, d_m .

Fröberg's conjecture is known to be true for

- $m \leq n$. In this case, semi-regular sequences are regular sequences.
- $n = 2$ [Frö85].
- $n = 3$. [Ani84]
- $m = n + 1$ over fields of characteristic 0 [Frö85].
- $d_1 = \dots = d_m = 2$ and $n \leq 11$, $d_1 = \dots = d_r = 3$ and $n \leq 8$ [FH02].

For an affine polynomial sequence, its regularity or semi-regularity is reflected on its homogeneous components of highest degrees.

3.2.5 Boolean semi-regular sequence

When a polynomial sequence are over the boolean field $\mathbb{F}_2 = \text{GaloisField}(2)$, it is called a boolean polynomial sequence. Since variables are all over \mathbb{F}_2 , relations $x_i^2 - x_i = 0$ hold for $i = 1, \dots, n$, which are commonly referred to as *field equations*. A boolean affine polynomial sequence is semi-regular if the sequence of homogeneous components of highest degree is semi-regular. Therefore, we consider a homogeneous sequence $(f_1, \dots, f_m) \in (\mathbb{F}_2[\mathbf{x}]/\langle x_1^2, \dots, x_n^2 \rangle)^m$.

Lemma 3.2.1 ([BFSY05]). Given polynomials $f_1, \dots, f_m \in \mathbb{F}_2[x_1, \dots, x_n]$, the index of regularity of a homogeneous ideal

$$I = \langle f_1, \dots, f_m, x_1^2, \dots, x_n^2 \rangle$$

in $\mathbb{F}_2[\mathbf{x}]$ is given by

$$i_{reg} = \min \left\{ d \geq 0 \mid \dim_{\mathbb{F}_2} \{f \in I, \deg(f) = d\} \cup \{0\} = \binom{n}{d} \right\}.$$

Definition 3.2.17 (Boolean homogeneous semi-regular sequence [BFSY05]). Given a homogeneous quadratic sequence $(f_1, \dots, f_m) \in (\mathbb{F}_2[\mathbf{x}]/\langle x_1^2, \dots, x_n^2 \rangle)^m$, it is called a *boolean homogeneous semi-regular sequence* if for all $i = 1, \dots, m$ and $g \in \mathbb{F}_2[\mathbf{x}]/\langle x_1^2, \dots, x_n^2 \rangle$ such that

$$gf_i \in \langle f_1, \dots, f_{i-1} \rangle \text{ and } \deg(gf_i) \leq i_{reg},$$

then g is also in $\langle f_1, \dots, f_{i-1}, f_i \rangle$.

Proposition 3.2.11 (Hilbert series of a boolean homogeneous semi-regular sequence [BFSY05]). Given a homogeneous semi-regular sequence

$$F = (f_1, \dots, f_m) \in (\mathbb{F}_2[\mathbf{x}]/\langle x_1^2, \dots, x_n^2 \rangle)^m,$$

its Hilbert series is given by

$$\left[\frac{(1+t)^n}{\prod_{i=1}^m (1+t^{d_i})} \right]_+,$$

where $[a]_+$ means the series obtained by truncating a at its first non-positive coefficient.

3.2.6 Weighted-homogeneous gradings and multi-homogeneous gradings

Similar to homogeneous gradings, we can generalize all concepts to graded algebra with weighted variables and partitioned variables. In a weighted-homogeneous grading case, a weight is attached to each variable. In a multi-homogeneous grading case, each variable belongs to a partition of variables.

Definition 3.2.18 (Homogeneous grading). Given $\mathbb{K}[\mathbf{x}]$, its *homogeneous grading* is given by the decomposition

$$\mathbb{K}[\mathbf{x}] = \bigoplus_{d=0}^{\infty} \mathbb{K}[\mathbf{x}]_d,$$

where $\mathbb{K}[\mathbf{x}]_d$ is the \mathbb{K} -vector space generated by monomials of degree d .

Similar definition can be given for *weighted-homogeneous grading*.

Definition 3.2.19 (Weighted-homogeneous grading). Given $\mathbf{w} = (w_1, \dots, w_n) \in \mathbb{N}^n$, the weighted degree of a monomial $x_1^{a_1} \cdots x_n^{a_n}$ is given by

$$wdeg_{\mathbf{w}}(x_1^{a_1} \cdots x_n^{a_n}) = \sum_{i=1}^n w_i a_i.$$

The *weighted-homogeneous grading* on $\mathbb{K}[\mathbf{x}]$, given the weight degree as \mathbf{w} , is given by the decomposition

$$\mathbb{K}[\mathbf{x}] = \bigoplus_{i=0}^{\infty} \mathbb{K}[\mathbf{x}]_{wi},$$

where $\mathbb{K}[\mathbf{x}]_{wi}$ is the \mathbb{K} -vector space generated by monomials of weighted degree i .

Definition 3.2.20 (Multi-homogeneous grading). Let $\mathbf{x}_1, \dots, \mathbf{x}_l$ be a partition of the variables in $\mathbb{K}[\mathbf{x}]$, hence we have

$$\mathbb{K}[\mathbf{x}] = \bigotimes_{i=1}^l \mathbb{K}[\mathbf{x}_i].$$

The multi-homogeneous grading on $\mathbb{K}[\mathbf{x}]$, with a partition on \mathbf{x} given by $\mathbf{x}_1, \dots, \mathbf{x}_l$, is given by

$$\mathbb{K}[\mathbf{x}] = \bigoplus_{(d_1, \dots, d_l) \in \mathbb{N}^l} \mathbb{K}[\mathbf{x}]_{(d_1, \dots, d_l)},$$

where $\mathbb{K}[\mathbf{x}]_{(d_1, \dots, d_l)}$ is the \mathbb{K} -vector space $\mathbb{K}[\mathbf{x}_1]_{d_1} \otimes \cdots \otimes \mathbb{K}[\mathbf{x}_l]_{d_l}$.

Definition 3.2.21 (Hilbert function and Hilbert series for a homogeneous grading).

Given $I \subset \mathbb{K}[\mathbf{x}]$ be a homogeneous ideal, its *Hilbert function* $HF_{\mathbb{K}[\mathbf{x}]/I}$ is given by

$$HF_{\mathbb{K}[\mathbf{x}]/I}(d) = \dim_{\mathbb{K}}(\mathbb{K}[\mathbf{x}]_d/I_d)$$

and its *Hilbert series* is given by

$$HS_{\mathbb{K}[\mathbf{x}]/I}(t) = \sum_{d=0}^{\infty} HF_{\mathbb{K}[\mathbf{x}]/I}(d)t^d.$$

Definition 3.2.22 (Hilbert function and Hilbert series for a weighted homogeneous grading). Given $I \subset \mathbb{K}[\mathbf{x}]$ be a homogeneous ideal with weights $\mathbf{w} \in \mathbb{N}^n$ added to its variables, then its *weighted Hilbert function* $wHF_{\mathbb{K}[\mathbf{x}]/I}$ is given by

$$wHF_{\mathbb{K}[\mathbf{x}]/I}(d) = \dim_{\mathbb{K}}^{\mathbf{w}}(\mathbb{K}[\mathbf{x}]_d/I_d)$$

and its *weighted Hilbert series* is given by

$$wHS_{\mathbb{K}[\mathbf{x}]/I}(t) = \sum_{d=0}^{\infty} wHF_{\mathbb{K}[\mathbf{x}]/I}(d)t^d.$$

Definition 3.2.23 (Hilbert function and Hilbert series for a multi-homogeneous grading). Given $I \subset \mathbb{K}[\mathbf{x}]$ be a multi-homogeneous ideal with a partition $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_l)$ on its variables, then its *multi-Hilbert function* $mHF_{\mathbb{K}[\mathbf{x}]/I}$ is given by

$$mHF_{\mathbb{K}[\mathbf{x}]/I}(d_1, \dots, d_l) = \dim_{\mathbb{K}}(\mathbb{K}[\mathbf{x}]_{\{d_1, \dots, d_l\}}/I_{\{d_1, \dots, d_l\}})$$

and its *multi-Hilbert series* is given by

$$mHS_{\mathbb{K}[\mathbf{x}]/I}(t_1, \dots, t_l) = \sum_{\mathbf{d} \in \mathbb{N}^l} mHF_{\mathbb{K}[\mathbf{x}]/I}(\mathbf{d})t_1^{d_1} \cdots t_l^{d_l}.$$

Example 3.2.2. For the graded ring $\mathbb{K}[\mathbf{x}]$, its Hilbert series is

$$HS_{\mathbb{K}[\mathbf{x}]}(t) = \frac{1}{(1-t)^n};$$

its *weighted Hilbert series*, given the added weights on variables \mathbf{x} being \mathbf{w} , is

$$wHS_{\mathbb{K}[\mathbf{x}]}^{\mathbf{w}}(t) = \frac{1}{\prod_{i=1}^n (1-t^{w_i})};$$

its *multi-Hilbert series*, given a partition $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_l)$, is

$$mHS_{\mathbb{K}[\mathbf{x}]}(t_1, \dots, t_l) = \frac{1}{\prod_{i=1}^l (1-t_i)^{m_i}},$$

where $m_i = |\mathbf{x}_i|$, the number of variables in \mathbf{x}_i .

3.3 Gröbner bases and algorithms

3.3.1 Monomial orderings

Given linear polynomials $l_1, \dots, l_m \in \mathbb{K}[x_1, \dots, x_n]$, mathematically we can consider the vector space generated by l_1, \dots, l_m . A echelonized basis for this vector space can be obtained by performing operations on l_1, \dots, l_m . When we are given non-linear polynomials $f_1, \dots, f_m \in \mathbb{K}[\mathbf{x}]$, mathematically we consider the ideal generated by f_1, \dots, f_m . A special basis with good property, Gröbner basis, can be obtained by performing operations on f_1, \dots, f_m , and monomial orderings are needed to be able to do that.

In Definition 3.1.1, monomials, coefficient, exponent, term and total degree are defined.

Definition 3.3.1 (Monomial ordering [CLO15]). A *monomial ordering* \succ on $\mathbb{K}[\mathbf{x}]$ is a total-ordering on the set of monomials $\mathbf{x}^{\mathbf{a}}, \mathbf{a} \in \mathbb{N}^n$, satisfying:

- \succ is a well-ordering on \mathbb{N}^n , which means there is the smallest element for any subset of \mathbb{N}^n under \succ .
- If $\mathbf{x}^{\mathbf{a}} \succ \mathbf{x}^{\mathbf{b}}, \mathbf{r} \in \mathbb{N}^n$, then $\mathbf{x}^{\mathbf{a}}\mathbf{x}^{\mathbf{r}} \succ \mathbf{x}^{\mathbf{b}}\mathbf{x}^{\mathbf{r}}$.

Given a monomial ordering, we are able to define the following notions:

Definition 3.3.2 (Leading monomial, leading term and leading coefficient). Given an element $f \in \mathbb{K}[\mathbf{x}]$ and let \succ be a monomial ordering. Then,

- the *leading monomial* of f with respect to \succ , $\text{LM}_{\succ}(f)$, is the biggest monomial that appears in f ,
- the *leading coefficient* of f with respect to \succ , $\text{LC}_{\succ}(f)$, is the coefficient associated to the leading monomial of f ,
- the *leading term* of f with respect to \succ , $\text{LT}_{\succ}(f)$, is the product of the leading monomial and leading coefficient of f , i.e. $\text{LT}_{\succ}(f) = \text{LC}_{\succ}(f) \cdot \text{LM}_{\succ}(f)$.

Definition 3.3.3 (Support and terms). Let $f \in \mathbb{K}[\mathbf{x}], f \neq 0$, i.e.

$$f = \sum c(a_1, \dots, a_n) x_1^{a_1} \cdots x_n^{a_n},$$

where $c(a_1, \dots, a_n) \in \mathbb{K}$. The *support* of f is defined by

$$\text{Support}(f) := \{x_1^{a_1} \cdots x_n^{a_n} \mid c(a_1, \dots, a_n) \neq 0\},$$

and the set of terms of f is given by

$$T(f) := \{c(a_1, \dots, a_n) \cdot x_1^{a_1} \cdots x_n^{a_n} \mid c(a_1, \dots, a_n) \neq 0\}.$$

Definition 3.3.4 (Initial ideal). Let $I \subset \mathbb{K}[\mathbf{x}]$ be an ideal of $\mathbb{K}[\mathbf{x}]$ with a monomial ordering \succ . All the leading monomials of polynomials in I can form an ideal, which we call the *initial ideal* of I .

$$I_{ini}(I) := \langle \text{LM}_\succ(f) \mid f \in I \rangle.$$

For the polynomial ring $\mathbb{K}[\mathbf{x}]$, several monomial orderings are commonly used, such as lexicographical ordering and graded reverse lexicographical ordering.

Definition 3.3.5 (Lexicographical ordering (lex)). Given a monomial ordering with $x_1 \succ x_2 \succ \cdots \succ x_n$, then the order \succ is called a *lexicographical ordering* (lex) if

$$\mathbf{x}^{\mathbf{a}} \succ \mathbf{x}^{\mathbf{b}} \iff a_i = b_i \ (\forall i < k), a_k \succ b_k \text{ for some } k.$$

Example 3.3.1 (Lex ordering). Consider the Lex ordering on $\mathbb{K}[x, y, z]$ with $x \succ y \succ z$, then $x^2 \succ y^4 \succ yz^5, x^2z^3 \succ x^2z^2$.

It is easy to verify that for an element $f \in \mathbb{K}[\mathbf{x}]$, if its leading term with respect to lexicographical order lies in $\mathbb{K}[x_i, \dots, x_n]$, then $f \in \mathbb{K}[x_i, \dots, x_n]$.

Definition 3.3.6 (Graded reverse lexicographical ordering (grevlex)). Given a monomial ordering with $x_1 \succ x_2 \succ \cdots \succ x_n$, then the order \succ is called a *graded reverse lexicographical ordering* (grevlex) if

$$\mathbf{x}^{\mathbf{a}} \succ \mathbf{x}^{\mathbf{b}} \iff \begin{aligned} & \sum_{i=1}^n a_i > \sum_{i=1}^n b_i \text{ or} \\ & \sum_{i=1}^n a_i = \sum_{i=1}^n b_i \text{ and } \exists k \text{ s.t. } \forall i > k \ a_i = b_i \text{ and } a_k < b_k. \end{aligned}$$

Even though the graded reverse lexicographical ordering is not very intuitive, but for some computation using this ordering is the most efficient. There are also other numerous monomial orderings, but only the orderings, such as lex and grevlex, with good computational properties are for practical use.

3.3.2 Gröbner bases and algorithms

Definition 3.3.7 (Gröbner basis). Let $I \in \mathbb{K}[\mathbf{x}]$ be an ideal of $\mathbb{K}[\mathbf{x}]$ with a monomial ordering \succ . A Gröbner basis with respect to \succ is a set $G = (g_1, \dots, g_s) \subset I$ such that

$$\langle \text{LT}_\succ(g_1), \dots, \text{LT}_\succ(g_s) \rangle = \langle \text{LT}_\succ(f) \mid f \in I \rangle.$$

Equivalently, a Gröbner basis of I can also be defined as a set $G \subset I$ such that $\forall f \in I, f \neq 0, \exists g \in G, \text{LT}_{\succ}(g) \mid \text{LT}_{\succ}(f)$. Namely, the leading term of any polynomials in I can be reduced to 0 by leading terms of G . More generally, any element in I can be reduced to 0 using G .

Proposition 3.3.1 ([CLO15], Ch. 2, Sec. 5). Given an ideal $I \subset \mathbb{K}[x_1, \dots, x_n]$ with a monomial order, then I has a Gröbner basis. Furthermore, any Gröbner basis for I serves as a generating set for I .

Definition 3.3.8 (Top-reducible). Let $f, g \in \mathbb{K}[\mathbf{x}] \setminus \{0\}$ with a monomial ordering \succ . f is said to be *top-reducible* by g with respect to \succ if $\text{LM}_{\succ}(g) \mid \text{LM}_{\succ}(f)$. More generally, given a set $G = (g_1, \dots, g_s) \in \mathbb{K}[\mathbf{x}]^s$ and $f \in \mathbb{K}[\mathbf{x}] \setminus \{0\}$, f is said to be *top-reducible* by G if $\exists g \in G, \text{LM}_{\succ}(g) \mid \text{LM}_{\succ}(f)$.

Algorithm 3.1 is an algorithm for polynomial reduction.

Algorithm 3.1: Polynomial reduction algorithm

Input : $f \in \mathbb{K}[\mathbf{x}]$, $G = \{g_1, \dots, g_s\} \in \mathbb{K}[\mathbf{x}]^s$, a monomial ordering \succ

Output: r s.t. r is not top-reducible by F and $f - r \in \langle G \rangle$

```

1  $i \leftarrow 0, r \leftarrow f$ 
2 while  $r \neq 0$  and  $i \leq s$  do
3    $i \leftarrow i + 1$ 
4   if  $r$  is top-reducible by  $f_i$  then
5      $r \leftarrow r - \frac{\text{LT}_{\succ}(r)}{\text{LT}_{\succ}(f_i)} f_i$ 
6      $i \leftarrow 0$ 
7 Return  $r$ .
```

Proposition 3.3.2 ([CLO15], Sec. 2.6, Prop. 1). Let $I \subset \mathbb{K}[\mathbf{x}]$ be an ideal and $G = \{g_1, \dots, g_s\}$ be a Gröbner basis for I . Then given $f \in \mathbb{K}[\mathbf{x}]$, there is a unique $r \in \mathbb{K}[\mathbf{x}]$ such that no monomial of r is divisible by any of $\text{LM}(g_1), \dots, \text{LM}(g_s)$ and $f - r \in I$.

Definition 3.3.9 (Normal form [CLO15], Ch. 2, Sec. 6). Let $I \in \mathbb{K}[\mathbf{x}]$ be an ideal and $G = \{g_1, \dots, g_s\}$ be a Gröbner basis for I . Then given $f \in \mathbb{K}[\mathbf{x}]$, there is a unique $r \in \mathbb{K}[\mathbf{x}]$ such that $\exists g \in I, f = g + r$ and no monomials of r is divisible by any of $\text{LM}(g_1), \dots, \text{LM}(g_s)$. The polynomial r is called the *normal form* of f modulo G , written

$$f \xrightarrow{G} r.$$

Algorithm 3.2 is an algorithm for computing the full polynomial reduction of every monomial of f on a set of polynomials G , that we call a *full reduction*. The result of this full reduction heavily relies on the order in G , different order yields different results. However, when G is a Gröbner basis, this result becomes unique, and is the normal form [CLO15].

Algorithm 3.2: Full polynomial reduction

Input : $f \in \mathbb{K}[\mathbf{x}]$, $G = \{g_1, \dots, g_s\} \in \mathbb{K}[\mathbf{x}]^s$, a monomial ordering \succ

Output: r s.t. no monomial of r is top-reducible by G and $f - r \in \langle G \rangle$.

```

1  $r \leftarrow 0, p \leftarrow f$ 
2 while  $p \neq 0$  do
3   if  $\exists g \in G$  s.t.  $p$  is top-reducible by  $g$  then
4      $p \leftarrow p - \frac{\text{LT}_\succ(p)}{\text{LT}_\succ(g)}g$ 
5   else
6      $r \leftarrow r + \text{LT}_\succ(p)$ 
7      $p \leftarrow \text{LT}_\succ(p)$ 
8 Return  $r$ .
```

Gröbner bases are not unique, as we can add any polynomials from an ideal to a Gröbner basis and still remain a Gröbner basis. However, under some more restrictions, Gröbner bases can become unique, which we call a reduced Gröbner basis.

Definition 3.3.10 (Reduced Gröbner basis [CLO15]). A Gröbner basis $G = (g_1, \dots, g_s)$ to an ideal I is called a reduced Gröbner basis if

- $\forall g \in G, \text{LC}_\succ(g) = 1$.
- $\forall g \in G, \nexists t \in T(g), t \in \langle \text{LM}_\succ(G \setminus \{g\}) \rangle$.

Proposition 3.3.3 ([CLO15]). Given a non-zero ideal I and a monomial ordering, there is a unique reduced Gröber basis for I .

3.3.3 Buchberger's algorithm

The key concept to Buchberger's algorithm [Buc65] is the S-polynomial.

Definition 3.3.11 (Least common multiple). Let $\mathbf{x}^{\mathbf{a}} = x_1^{a_1} \cdots x_n^{a_n}$ and $\mathbf{x}^{\mathbf{b}} = x_1^{b_1} \cdots x_n^{b_n}$ be two monomials in $\mathbb{K}[\mathbf{x}]$, then their *least common multiple* $\text{LCM}(\mathbf{x}^{\mathbf{a}}, \mathbf{x}^{\mathbf{b}})$ is defined by

$$\text{LCM}(\mathbf{x}^{\mathbf{a}}, \mathbf{x}^{\mathbf{b}}) := \prod_{i=1}^n x_i^{\max(a_i, b_i)}.$$

In fact, the ideal generated by monomials in $\text{LCM}(\mathbf{x}^{\mathbf{a}}, \mathbf{x}^{\mathbf{b}})$ is the intersection of $\langle \mathbf{x}^{\mathbf{a}} \rangle$ and $\langle \mathbf{x}^{\mathbf{b}} \rangle$.

Definition 3.3.12 (S-polynomial). Given $f, g \in \mathbb{K}[\mathbf{x}]$ with a monomial ordering \succ , the *S-polynomial* of f and g with respect to \succ is defined by

$$S_{\text{poly}}(f, g) := \text{LCM}(\text{LM}(f), \text{LM}(g)) \left(\frac{f}{\text{LT}(g)} - \frac{g}{\text{LT}(f)} \right).$$

Buchberger's algorithm starts with a list of *critical pairs*, which are pairs of polynomials to be examined, and it computes full reduction of every S-polynomial associated to every critical pair, then add new critical pairs accounted for a non-zero remainder if exists until all critical pairs are examined. The detailed procedures are shown in Algorithm 3.3.

The termination and correctness of Algorithm 3.3 relies on Buchberger's criterion.

Theorem 3.3.1 (Buchberger's Criterion [CLO15], Ch. 2, Sec. 6, Th. 6). Given an ideal $I \subset \mathbb{K}[\mathbf{x}]$ and a monomial ordering, a basis $G = \{g_1, \dots, g_s\}$ of I is a Gröbner basis of I if and only if all *S-polynomials* associated to all pairs $\{(g_i, g_j) \mid g_i, g_j \in G, i > j\}$ reduce to 0 by G .

Theorem 3.3.2 (Termination of Buchberger's Algorithm [CLO15], Ch. 2, Sec. 7, Th. 2). Given an ideal $I \subset \mathbb{K}[\mathbf{x}]$, then a Gröbner basis can be computed in a finite number of steps with Algorithm 3.3.

In this very first version of Buchberger's algorithm, many pairs are examined, among which there are cases that S-polynomials of some pairs get reduced to 0 during the whole computation process. This does not contribute in finding useful polynomials

Algorithm 3.3: Buchberger's algorithm

Input : $F = (f_1, \dots, f_s) \in \mathbb{K}[\mathbf{x}]^s$, \succ **Output:** A Gröbner basis of $\langle f_1, \dots, f_s \rangle$

```

1  $G \leftarrow F$ 
2  $C \leftarrow \{(f_i, f_j) \mid f_i, f_j \in F, i > j\}$ 
3 while  $C \neq \emptyset$  do
4   pick  $(f, g) \in C, C \leftarrow C \setminus \{(f, g)\}$ 
5    $h \leftarrow S_{poly}(f, g)$ 
6    $h \leftarrow \mathbf{Full\ reduction}(h, G)$ 
7   if  $h \neq 0$  then
8      $C \leftarrow C \cup \{(f_i, h) \mid f_i \in G\}$ 
9      $G \leftarrow G \cup \{h\}$ 
10 Return  $G$ 

```

for constructing a Gröbner basis. Therefore, pairs selecting strategies are essential in improving the performances of this type of algorithms.

Another type of algorithms for computing Gröbner basis, developed by Jean-Charles Faugère select many pairs together and use linear algebra techniques to perform reduction on many S-polynomials together. The F4 algorithm [Fau99] and the F5 algorithm [Fau02] fall into this category. Moreover, complexities of this type of algorithms are also easier to estimate since most complexity-costly computations come from row operations on matrices.

Example 3.3.2 (Buchberger's algorithm). Compute a Gröbner basis for $\{f_1 = 2xy + y + 2, f_2 = 2xy + 2x + y^2 + 2y\} \subset \mathbb{F}_3[x, y]$ with a lex monomial ordering on $x \succ y$.

1. $G \leftarrow \{f_1, f_2\}$
2. $S_{poly}(f_1, f_2) \xrightarrow{G} 2x + y^2 + y + 1, f_3 := 2x + y^2 + y + 1, G \leftarrow G \cup \{f_3\}$
3. $S_{poly}(f_1, f_3) \xrightarrow{G} y^3 + y^2 + 1, f_4 := y^3 + y^2 + 1, G \leftarrow G \cup \{f_4\}$
4. $S_{poly}(f_i, f_4) \xrightarrow{G} 0$ for $i = 1, 2, 3$.
5. Obtain a Gröbner basis $\{f_1, f_2, f_3, f_4\}$.

3.3.4 F4 algorithm

The F4 algorithm [Fau99] has a very similar rationale to Buchberger's algorithm, but it improves the complexity-costly polynomial reduction by using sparse linear algebra techniques. The F4 algorithm reduces many critical pairs at the same time by putting their associated elements needed for polynomial reduction in a large matrix and reduce the matrix to a row echelon form. Algorithm 3.4 gives an outline of the F4 algorithm.

Algorithm 3.4: The F4 algorithm

Input : $F = (f_1, \dots, f_s) \in \mathbb{K}[\mathbf{x}]^s$, \succ
Output: A Gröbner basis of $\langle f_1, \dots, f_s \rangle$

```

1  $G \leftarrow F$ 
2  $P \leftarrow \left\{ \left( \frac{\text{LCM}(\text{LM}(g_i), \text{LM}(g_j))}{\text{LM}(g_i)} g_i, \frac{\text{LCM}(\text{LM}(g_i), \text{LM}(g_j))}{\text{LM}(g_j)} g_j \right) \mid g_i, g_j \in G, i > j \right\}$ 
3  $d \leftarrow 0$ 
4 while  $P \neq \emptyset$  do
5    $d \leftarrow d + 1$ 
6    $P_d \leftarrow \text{Select}(P), P \leftarrow P \setminus P_d$ 
7    $L_d \leftarrow \{a, b \mid (a, b) \in P_d\}$ 
8    $L_d \leftarrow \text{SymbolicPreprocessing}(L_d, G)$ 
9    $F_d \leftarrow \text{Reduction}(L_d, G)$ 
10  for  $h \in F_d$  do
11    if  $\text{LM}(h) \notin \text{LM}(G)$  then
12       $P \leftarrow \left\{ \left( \frac{\text{LCM}(\text{LM}(g_i), \text{LM}(h))}{\text{LM}(g_i)} g_i, \frac{\text{LCM}(\text{LM}(g_i), \text{LM}(h))}{\text{LM}(h)} h \right) \mid g_i \in G \right\}$ 
13       $G \leftarrow G \cup \{h\}$ 
14 Return  $G$ 

```

In this algorithm, there are three subroutines used, **Select**, **SymbolicPreprocessing** and **Reduction**. The function **Select** chooses many critical pairs for examination, and additional selective strategy can be applied in this function to avoid zero reduction that does not contribute to constructing a Gröbner basis. The function **SymbolicPreprocessing** can be considered as a preparation step for polynomial reduction in the F4 algorithm. To better explain the role of this function, we first take a

look at an example.

Example 3.3.3 (Polynomial reduction and linear algebra). Given $f = x^y + 3xy + 2y^3 \in \mathbb{F}_3[x, y]$ with a lex monomial ordering on $x \succ y$, compute a full reduction of f on the set $\{g_1 = x^2 + y, g_2 = y + 2\}$, that is to reduce all monomials $M = \{x^2y, xy, y^3\}$ of f .

1. $M \leftarrow M \setminus \{x^2y\}$, $x^2y - yg_1 = -y^2$,
 y^2 is top-reducible by G , $M \leftarrow M \cup \{y^2\} = \{xy, y^3, y^2\}$
2. $M \leftarrow M \setminus \{xy\}$, $xy - xg_2 = -2x$,
 x is not top-reducible by G , $M = \{y^3, y^2\}$
3. $M \leftarrow M \setminus \{y^3\}$, $y^3 - y^2g_2 = -2y^2$,
 y^2 is already in M , $M = \{y^2\}$
4. $M \leftarrow M \setminus \{y^2\}$, $y^2 - yg_2 = -2y$,
 y is top-reducible by G , $M \leftarrow M \cup \{y\} = \{y\}$
5. $M \leftarrow M \setminus \{y\}$, $y - g_2 = -2$
 1 is not top-reducible by G , $M = \{0\}$.

To reduce f by G to -2 eventually, we need to subtract $\{yg_1, xg_2, y^2g_2, yg_2, g_2\}$ from f . Since the action of subtraction on polynomials is analogous to row operations on a matrix, we are able to perform polynomial reduction on f with G using additional polynomials $\{yg_1, xg_2, y^2g_2, yg_2, g_2\}$ by extracting all coefficients of every polynomial, putting them in a large matrix and performing row operations. This is also the rationale behind the function **SymbolicPreprocessing** and **Reduction**.

Definition 3.3.13 (Coefficient matrix). Given $F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$, let $T = (t_1, \dots, t_r)$ be the list of monomials appeared in F sorted descending with respect to a monomial order \succ , and $C(f_i, t_j)$ be the coefficient of the term that contains t_j in f_i . Then the *coefficient matrix* of F is defined as

$$\mathcal{C}(F) := [C(f_i, t_j)] \in \mathbb{K}^{m \times r}.$$

Each row of the *coefficient matrix* represents a polynomial $f_i = \sum_{j=1}^r C(f_i, t_j)t_j$ for $i = 1, \dots, m$. Performing row operations on a *coefficient matrix* is equivalent to performing addition or subtraction on polynomials associated to the matrix. From a reduced echelon form of a *coefficient matrix*, we can obtain a set of new polynomials and they lie in the ideal generated by f_1, \dots, f_m .

Algorithm 3.5: Symbolic preprocessing

Input : $L = (l_1, \dots, l_t), G = (g_1, \dots, g_s) \in \mathbb{K}[\mathbf{x}]^s, \succ$ **Output:** A set of polynomials

```

1  $E \leftarrow \{\}$ 
2  $M \leftarrow T(L)$ 
3 while  $E \neq M$  do
4    $m \leftarrow$  largest monomial in  $M \setminus E$ 
5    $E \leftarrow E \cup \{m\}$ 
6   if  $\exists g \in G$  s.t.  $LM(g) \mid m$  then
7      $L \leftarrow L \cup \left\{ g \frac{m}{LM(g)} \right\}$ 
8 Return  $L$ 

```

Algorithm 3.6: Reduction

Input : $L = (l_1, \dots, l_t), G = (g_1, \dots, g_s) \in \mathbb{K}[\mathbf{x}]^s, \succ$ **Output:** A set of polynomials

```

1  $C \leftarrow \mathcal{C}(L)$ 
2  $C' \leftarrow$  reduced echelon form of  $C$ 
3  $L' \leftarrow$  the set of polynomials corresponding to the rows of  $C'$ 
4  $G' \leftarrow \{f \mid f \in L', LM(f) \notin LM(G)\}$ 
5 Return  $G'$ 

```

To understand the complexity of computing a Gröbner basis using the F4 algorithm, a notion called *degree of regularity* is needed.

Definition 3.3.14 (Maucaulay matrix). Given $F = \{f_1, \dots, f_m\} \in \mathbb{K}[\mathbf{x}]$ of respective degrees d_1, \dots, d_m with a monomial ordering \succ . For $d \in \mathbb{N}$, let M_d be the set of monomials in $\mathbb{K}[\mathbf{x}]$ of degree no larger than d sorted decreasingly, there are in total $\binom{n+d}{d}$ such monomials and define a set of polynomials

$$N := \cup_{i=1}^m \{f_i m_{d-d_i} \mid m_{d-d_i} \in M_{d-d_i}\},$$

which has

$$\sum_{i=1}^m \binom{n+d-d_i}{d-d_i}$$

elements. Then the coefficient matrix of N of size

$$\sum_{i=1}^m \binom{n+d-d_i}{d-d_i} \times \binom{n+d}{d}$$

with respect to M_d is called the *Macaulay matrix* of f_1, \dots, f_m in degree d , which we denote it by $\mathcal{M}_d(F)$.

Theorem 3.3.3 ([Laz83]). Let $F = \{f_1, \dots, f_m\} \in \mathbb{K}[\mathbf{x}]$, there exists a degree d such that the row echelon form of $\mathcal{M}_d(F)$ form a Gröbner basis of the ideal generated by F .

From this proposition, an algorithm can be formed for computing Gröbner basis, and its complexity is bounded by performing row echelon form reduction on a Macaulay matrix of degree d . This degree is commonly known as *degree of regularity*.

Definition 3.2.12 and 3.2.11 give definitions on the degree of an ideal and the homogeneous index of regularity of a homogeneous ideal. For a 0-dimensional homogeneous ideal I ,

$$i_{reg} = \deg(HS_{\mathbb{K}[\mathbf{x}] \setminus I}) + 1$$

since for a homogeneous ideal, its Hilbert series is given by

$$HS_{\mathbb{K}[\mathbf{x}] \setminus I}(t) = \frac{N(t)}{(1-t)^{\dim(I)}},$$

which turns into a polynomial given $\dim(I) = 0$. Therefore, the homogeneous index of regularity can be read off from the Hilbert series in the 0-dimensional homogeneous case. Moreover, $i_{reg}(I)$ bounds the degree of all polynomials in a reduced homogeneous Gröbner basis of I . In the affine 0-dimensional cases, we consider the generalization of index of regularity, which is *degree of regularity*.

Definition 3.3.15 (Degree of regularity). Let $F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$ and $\tilde{F} = (\tilde{f}_1, \dots, \tilde{f}_m)$ be the homogeneous components of the highest degrees polynomials in F . If $\dim(\langle \tilde{F} \rangle) = 0$, then $\dim(\langle F \rangle) = 0$ and we refer to the index of regularity of $\langle \tilde{F} \rangle$ as the degree of regularity of F , denoted by $d_{reg}(F)$ or $d_{reg}(\langle F \rangle)$.

For 0-dimensional homogeneous polynomial system, since there is no degree falls happening during a run of Gröbner bases computation, its complexity is bounded by performing linear algebra techniques on a large Macaulay matrix of size $N_{rows} \times N_{cols}$ of rank r , and this complexity is bounded by

$$O(N_{rows} \cdot N_{cols} \cdot r^{\omega-2}) [\text{Sto00}], \quad (3.1)$$

where $2 < \omega \leq 3$ is the linear algebra constant. Bounds for ω are given as:

- $\omega \leq 3$: schoolbook matrix multiplication;
- $\omega \leq 2.807$: Strassen's algorithm; [Str69]
- $\omega \leq 2.376$: Coppersmith-Winograd's algorithm [CW90];
- $\omega \leq 2.373$: Improvements on Coppersmith-Winograd's algorithm [Wil12].

For degree i , we have the size of a Macaulay matrix as

$$\text{Nrow}_i = \sum_{i=1}^m \binom{n+i-\deg(f_i)-1}{i-\deg(f_i)},$$

$$\text{Ncol}_i = \binom{n+i-1}{i},$$

$$r_i = \binom{n+i-1}{i} - HF_{\mathbb{K}[\mathbf{x}] \setminus \langle F \rangle}(i).$$

Therefore, computing the row echelon forms up to degree $d_{reg}(F)$ requires a complexity of

$$\begin{aligned} O \left(\sum_{i=0}^{d_{reg}(F)} (\text{Nrow}_i \cdot \text{Ncol}_i \cdot r^{\omega-2}) \right) &\leq O \left(\sum_{i=0}^{d_{reg}(F)} m \binom{n+i-1}{i}^{\omega} \right) \\ &\leq O \left(m \binom{n+d_{reg}(F)}{d_{reg}(F)}^{\omega} \right). \end{aligned}$$

Similarly, for affine polynomial systems, if the corresponding system of homogeneous components of highest degrees is 0-dimensional, then the complexity of computing a Gröbner basis for the graded reverse lexicographical order is bounded by

$$O \left(m \binom{n+d_{reg}(F)}{d_{reg}(F)}^{\omega} \right).$$

3.3.5 Relation between notions of regularity

Let $F = (f_1, \dots, f_m) \in \mathbb{K}[\mathbf{x}]^m$ be an affine polynomial system, F^h be its homogenization and \tilde{F} be the system of its homogeneous components of highest degrees. Let G (resp. G^h , \tilde{G}) be a minimal graded reverse lexicographical ordering Gröbner basis of the

ideal generated by F (resp. F^h, \tilde{F}). Since G can be obtained from G^h and dehomogenization, \tilde{G} can be obtained from evaluating the homogenization variable in G^h at 0. Regarding the maximal degrees in G, G^h and \tilde{G} , we have

$$\begin{aligned}\max\{\deg(g) \mid g \in G\} &\leq \max\{\deg(g^h) \mid g^h \in G^h\}, \\ \max\{\deg(\tilde{g}) \mid \tilde{g} \in \tilde{G}\} &\leq \max\{\deg(g^h) \mid g^h \in G^h\}.\end{aligned}$$

3.4 Solving polynomial systems

In the previous sections, we have recalled some basics about computing Gröbner basis, which can be used in solving polynomial systems. In this section, we recall some other algorithms and strategies for solving polynomial systems.

3.4.1 eXtended linearization (XL)

The eXtended linearization (XL) algorithm [CKPS00], similar to Gröbner basis techniques, is a general tool for solving systems of non-linear equations. It is especially efficient for overdetermined polynomial systems, but not efficient for determined or underdetermined polynomial systems compared with Gröbner basis techniques. There also exist some variations of XL algorithm such as FXL, XL2 and MutantXL [Moh11] for solving some specific polynomial systems. Moreover, studies in [AFI⁺04] show solving a polynomial system using XL algorithm is equivalent to calculate the reduced Gröbner basis of the ideal associated with the system, and it can be regarded as a redundant variant of the F4 algorithm. XL algorithm can also be coupled with the block Wiedemann algorithm [Tho02, Cop94, Wie86] as a linear algebra solver when matrices appeared in the XL algorithm are sparse. Computations in XL can also be easily parallelized [cCNY12, Moh11, TCS15]. The Wiedemann algorithm enables solving an $N \times N$ non-singular system with row sparsity k in $O(kN^2)$. When we perform the XL algorithm at degree d_{reg} , the polynomial system is expected to be solved, hence

$$N = \binom{n + d_{reg}}{d_{reg}}.$$

Since all rows of matrices in the XL algorithm are multiplication of a quadratic matrix with a monomial, there should be $\binom{n+2}{2}$ terms at most. Therefore, the complexity of

the XL algorithm using Wiedemann algorithm is

$$O\left(\binom{n+2}{n} \cdot \binom{n+d_{reg}}{d_{reg}}^2\right). \quad (3.2)$$

More specific complexity analysis of the XL algorithm can be found in [YC04]. The details of the XL algorithm is shown in Algorithm 3.7.

Algorithm 3.7: XL algorithm

Input : $F = (f_1, \dots, f_m) \in \mathbb{K}[x_1, \dots, x_n]^m$ of degree $d_1, \dots, d_m \in \mathbb{N}$, a monomial ordering \succ that sorts univariate monomials to the last

Output: A solution of $f_1 = \dots = f_m = 0$

```

1  $D \leftarrow \max_{i=1}^m d_i$ 
2 while  $D \geq 0$  do
3    $P_D \leftarrow \{\}$ 
4   for  $i \leftarrow 1$  to  $m$  do
5      $M_{D-d_i} \leftarrow$  Monomials of degree  $\leq D - d_i$ 
6      $P_D \leftarrow P_D \cup \{a \cdot f_i \mid a \in M_{D-d_i}\}$ 
7    $\text{sort}_{\succ}(P_D)$ 
8    $\hat{P}_D \leftarrow \text{Gaussian Elimination}(\text{linearization}(P_D))$ 
9   if  $\exists$  univariate polynomials  $\hat{f}$  in  $x_n$  then
10     $x_n = b_n \leftarrow$  Solve  $\hat{f}$  with Berlekamp's algorithm
11     $(x_1, \dots, x_n) = (b_1, \dots, b_n) \leftarrow \mathbf{XL}(F, x_n - b_n)$ 
12    Return  $(b_1, \dots, b_n)$ 
13  else
14     $D \leftarrow D + 1$ 
15  continue
```

3.4.2 Solving strategies for Gröbner basis techniques

In section 3.3, we recalled the definition of Gröbner bases and two algorithms for computing Gröbner bases. For Gröbner bases in lexicographical ordering, we have the following definition.

Definition 3.4.1 (Shape position). Given a reduced Gröbner basis of an ideal $I \subset \mathbb{K}[\mathbf{x}]$ in lexicographical monomial ordering, then we say I is in shape position if G has the following shape:

$$\begin{cases} g_1(x_1, \dots, x_n) = x_1 + r_1(x_n), \\ g_2(x_2, \dots, x_n) = x_2 + r_2(x_n), \\ \vdots \\ g_{n-1}(x_{n-1}, x_n) = x_{n-1} + r_{n-1}(x_n), \\ g_n(x_n) = r_1(x_n). \end{cases} \quad (3.3)$$

If an ideal I is in shape position, then it is also 0-dimensional, and for any 0-dimensional ideal, it is in shape position [GM05], this allows us to obtain solutions of a polynomial system through computing Gröbner basis in lexicographical monomial ordering.

Hybrid approach Often when solving a set of polynomials, Gröbner basis techniques are combined with exhaustive search, this approach is called the *hybrid approach* [BFP09, BFP12]. This approach searches for a best trade-off between Gröbner basis techniques and exhaustive search. It specifies a few variables in a polynomial system before computing a Gröbner basis. If it fails obtaining a solution, it respecifies those variables and repeat the same operations until a correct solution is found. [BFP09] claimed when $q \leq 2^{0.62\omega n}$, the hybrid approach is beneficial, where $2 < \omega \leq 3$ is the linear algebra constant.

Moreover, when a given polynomial system has many solutions and only one single solution is desired, a few variables can be specified to fasten the process of computing Gröbner bases and reduce the number of solutions. This is especially used in attacking multivariate signature cryptosystems algebraically since the public key polynomial map is surjective.

3.4.3 Solving overdetermined polynomial system

When a polynomial system has more polynomials than variables, it is called an overdetermined polynomial system. Let F be an overdetermined polynomial system with m polynomials in n variables. When $m \approx n$, its solving complexity is exponential, but when m increases to $\approx n^2$, there exists a polynomial time algorithm for solving it, which is *linearization*.

Quadratic polynomial in n variables has at most $\binom{n}{2}$ quadratic monomials, n linear

monomials and a constant. Linearization interprets each monomial as a new variables, and solves the new linear system using linear algebra techniques. When $m \geq \binom{n}{2} + n$, it expects to have only one solution. This method requires a complexity of

$$O(n^{2\omega}),$$

where $2 < \omega \leq 3$ is the linear algebra constant.

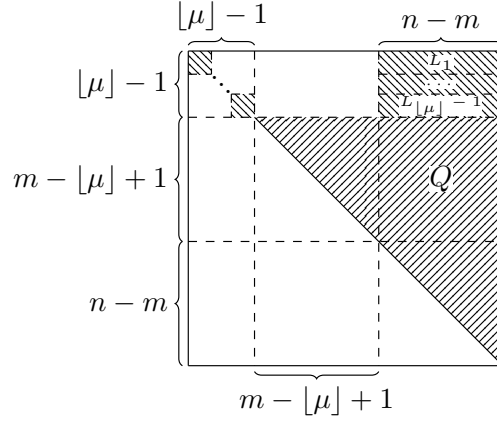
3.4.4 Solving underdetermined polynomial system

When a polynomial system has more variables than polynomials, it is called an underdetermined polynomial system. When $m \approx n$, its solving complexity is exponential, similar to the overdetermined case, when n increases to $\approx m^2$, there exists a polynomial algorithm [MHT13] with complexity $O(n^\omega m)$, where $2 < \omega \leq 3$ is the linear algebra constant. In [KPG03], Kipnis et al. showed an algorithm for solving polynomial systems under $n \geq m(m+1)$ over finite fields of even characteristic. Later in [CGMT02, TW12] extended the results in [KPG03] and showed the complexity of solving a homogeneous quadratic polynomial system of n polynomials in $n = \mu m$ variables is equivalent to solving a quadratic polynomial system of $m - \lfloor \mu \rfloor + 1$ equations and variables. We recall this solving method in this section.

Given a quadratic polynomial system $P = (p_1, \dots, p_m) \in \mathbb{F}_q[x_1, \dots, x_n]^m$, this method consists of two steps. First it finds a linear map $S : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ that transforms P into another polynomial system $F = P \circ S^{-1}$ such that polynomials in F are given by

$$f_i = \sum_{j=1}^{\lfloor \mu \rfloor - 1} a_j^{(i)} x_j^2 + \sum_{j=1}^{\lfloor \mu \rfloor - 1} x_j \cdot L_j^{(i)}(x_m, \dots, x_n) + Q^{(i)}(x_{\lfloor \mu \rfloor}, \dots, x_n), \quad (i = 1, \dots, m), \quad (3.4)$$

where $L_j(x_m, \dots, x_n)$ are linear polynomials in variables x_m, \dots, x_n and $Q(x_{\lfloor \mu \rfloor}, \dots, x_n)$ are quadratic polynomials in variables $x_{\lfloor \mu \rfloor}, \dots, x_n$. Matrix representation of polynomials in F is illustrated in Figure 3.1. Then we invert S and F to find a solution. More detailed descriptions are as follows.

Figure 3.1: Matrix representation of f_1, \dots, f_m

Step 1. Define matrices S_l of $n \times n$ for $l = 2, \dots, m$ given by

$$S_l := \begin{pmatrix} 1 & 0 & \dots & 0 & s_{1,l} & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & s_{2,l} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & s_{l-1,l} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & s_{l,l} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & s_{l+1,l} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & s_{n,l} & 0 & \dots & 1 \end{pmatrix}, \quad (3.5)$$

for $l = 2, \dots, m$ we recursively compute $P \circ S_2 \circ \dots \circ S_l$. Let $k = \min\{\lceil\mu\rceil - 1, l - 1\}$, for each S_l , the coefficients of $x_1 x_l, \dots, x_k x_l$ are set to zero in $P \circ S_2 \circ \dots \circ S_l$, which allows solving $s_{1,l}, \dots, s_{n,l}$ in S_l . These computations allow us to obtain a linear map $S_2 \circ \dots \circ S_m$ such that $P \circ S_2 \circ \dots \circ S_m$ are in the form of Figure 3.1. The detailed algorithm is shown in Algorithm 3.8.

Step 2. In this step, we solve the polynomial system P via map F and S . Inverting the map S is very easy and we focus on inverting the map F . We first let all linear polynomial in the upper right block in Figure 3.1 vanish, this gives us $m(\lceil\mu\rceil - 1)$ linear equations in $n - m$ variables, which expects to have one solution. Substituting this solution in f_1, \dots, f_m gives $\hat{f}_1, \dots, \hat{f}_m$, which have a matrix representation shown in Figure 3.2. More specifically, they are given by

$$\hat{f}_i = a_1^{(i)} x_1^2 + \dots + a_{\lceil\mu\rceil-1}^{(i)} x_{\lceil\mu\rceil-1}^2 + \hat{Q}^{(i)}(x_{\lceil\mu\rceil}, \dots, x_m), \quad (3.6)$$

Algorithm 3.8: Decompose P into $F = P \circ S_2 \circ \dots \circ S_m$

Input : $P = (p_1, \dots, p_m) \in \mathbb{K}[x_1, \dots, x_n]^m$ such that $n = \mu m$. ($\mu > 2$)

Output: A polynomial map F and a linear map S such that $P = F \circ S$

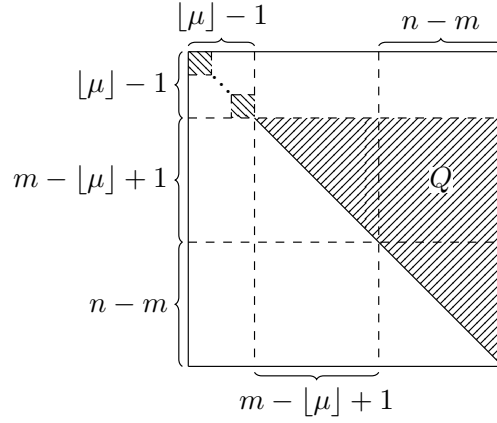
```

1 for  $i \leftarrow 2$  to  $m$  do
2    $S_l \leftarrow \begin{pmatrix} 1 & 0 & \dots & 0 & s_{1,l} & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 & s_{2,l} & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 & s_{l-1,l} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & s_{l,l} & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 & s_{l+1,l} & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & s_{n,l} & 0 & \dots & 1 \end{pmatrix}$ 
3    $k \leftarrow \min\{\lfloor \mu \rfloor - 1, l - 1\}$ 
4    $F_l \leftarrow P \circ S_l$ 
5    $Sys \leftarrow$  A linear system from coefficients of  $x_1 x_l, \dots, x_k x_l$  vanishing.
6    $S_l \leftarrow$  a solution of  $Sys$ 
7    $P \leftarrow P \circ S_l$ 
8  $F \leftarrow P$ 
9  $S \leftarrow (S_2 \circ \dots \circ S_m)^{-1}$ 
10 Return  $F, S$ .
```

after performing Gaussian elimination on $\hat{f}_1, \dots, \hat{f}_m$, we obtain $\bar{f}_1, \dots, \bar{f}_m$. The first $\lfloor \mu \rfloor - 1$ polynomials are given by:

$$\begin{aligned}
 x_1^2 &= \hat{Q}^{(1)}(x_{\lfloor \mu \rfloor}, \dots, x_m), \\
 &\vdots \\
 x_{\lfloor \mu \rfloor - 1}^2 &= \hat{Q}^{[\mu] - 1}(x_{\lfloor \mu \rfloor}, \dots, x_m).
 \end{aligned} \tag{3.7}$$

We substitute those expressions into the last $m - \lfloor \mu \rfloor + 1$ polynomial $\bar{f}_{\lfloor \mu \rfloor}, \dots, \bar{f}_m$ and obtain a multivariate quadratic polynomial system of $m - \lfloor \mu \rfloor + 1$ polynomials in $m - \lfloor \mu \rfloor + 1$ variables. This polynomial system can be solved by Gröbner basis techniques for example. A detailed description of this algorithm is shown in Algorithm 3.9

Figure 3.2: Matrix representation of $\hat{f}_1, \dots, \hat{f}_m$ **Algorithm 3.9:** Inversion of the polynomial map F

Input : $F = (f_1, \dots, f_m) \in \mathbb{K}[x_1, \dots, x_n]^m$ such that $n = \mu m$. ($\mu > 2$) and
 $f_i = \sum_{j=1}^{\lfloor \mu \rfloor - 1} a_j^{(i)} x_j^2 + \sum_{j=1}^{\lfloor \mu \rfloor - 1} x_j \cdot L_j^{(i)}(x_m, \dots, x_n) +$
 $Q^{(i)}(x_{\lfloor \mu \rfloor}, \dots, x_n), \quad (i = 1, \dots, m)$

Output: A solution of F

- 1 $(x_{m+1}, \dots, x_n) = (s_{m+1}, \dots, s_n) \in \mathbb{F}_q^{n-m} \leftarrow \left\{ L_1^{(i)} = 0, \dots, L_{\lfloor \mu \rfloor - 1}^{(i)} = 0 \right\}$ for
 $i = 1, \dots, m$.
- 2 $\hat{F} \leftarrow F(x_1, \dots, x_m, s_{m+1}, \dots, s_n)$
- 3 $\bar{F} = (\bar{f}_1, \dots, \bar{f}_{\lfloor \mu \rfloor - 1}, \bar{f}_{\lfloor \mu \rfloor}, \dots, \bar{f}_m) \leftarrow \text{GaussianElimination}(\hat{F})$

$$\bar{f}_1 = x_1^2 - \hat{Q}^{(1)}(x_{\lfloor \mu \rfloor}, \dots, x_m),$$

$$\vdots$$

$$\bar{f}_{\lfloor \mu \rfloor - 1} = x_{\lfloor \mu \rfloor - 1}^2 - \hat{Q}^{(\lfloor \mu \rfloor - 1)}(x_{\lfloor \mu \rfloor}, \dots, x_m).$$

- 4 $F' = (f'_{\lfloor \mu \rfloor}, \dots, f'_m) \leftarrow$ substitute the expressions of $x_1^2, \dots, x_{\lfloor \mu \rfloor - 1}^2$ into
 $\bar{f}_{\lfloor \mu \rfloor}, \dots, \bar{f}_m$.
- 5 $(x_{\lfloor \mu \rfloor}, \dots, x_m) = (a_{\lfloor \mu \rfloor}, \dots, a_m) \leftarrow$ solve F' with Gröbner basis techniques or XL
algorithm.
- 6 $(x_1, \dots, x_{\lfloor \mu \rfloor - 1}) = (a_1, \dots, a_{\lfloor \mu \rfloor - 1}) \leftarrow$ substitute $(a_{\lfloor \mu \rfloor}, \dots, a_m)$ for $(x_{\lfloor \mu \rfloor}, \dots, x_m)$
in \bar{F} .
- 7 $\mathbf{a} = (a_1, \dots, a_n)$
- 8 **Return** \mathbf{a} .

Chapter 4

Evaluating the security of EFC using algebraic techniques

At PQCrypto 2016, Szepieniec et al. proposed a new type of trapdoor called Extension Field Cancellation (EFC) for constructing secure multivariate encryption cryptosystems. They also specifically suggested two schemes EFC_p^- and $\text{EFC}_{pt^2}^-$ that apply this trapdoor and some modifiers. Although both of them seem to avoid all attacks used for cryptanalysis on multivariate cryptography, their decryption efficiency has room for improvement. On the other hand, their security was analyzed mainly through an algebraic attack of computing the Gröbner basis of the public key, and there possibly exists more effective attacks.

In this chapter, we introduce a more efficient decryption approach for EFC_p^- and $\text{EFC}_{pt^2}^-$, which manages to avoid all redundant computation involved in the original decryption algorithms without altering their public key. In addition, we estimate the secure parameters for EFC_p^- and $\text{EFC}_{pt^2}^-$ through a hybrid attack of algebraic attack and exhaustive search.

The result in this chapter was published on *IEICE transactions on Fundamentals of Electronics, Communication and Computer Sciences* under the title “The secure parameters and efficient decryption algorithm for multivariate public key cryptosystem EFC” [WIDT19].

4.1 Introduction

Ever since Shor [Sho97] introduced a polynomial-time algorithm in 1994 for solving the integer factorization problem and the discrete logarithm problem on quantum computers, on which currently used public key cryptosystems such as RSA and ECC are based, cryptology community has been researching on finding alternative cryptosystems that are quantum resistant (Post-quantum cryptography). Specially, the National Institute of Standards and Technology (NIST) in the United States is calling for post-quantum cryptosystems (PQC) proposals to be standardized, the National Security Agency (NSA) also announced their plan for switching to quantum resistant public key cryptosystems in the future.

Multivariate cryptography is considered as one of the main candidates for post-quantum cryptography because the security of multivariate cryptography is based on the hardness of the problem of solving a set of multivariate quadratic polynomials, which was proven to be NP-complete, and multivariate cryptography is in general very efficient and requires very modest computational resources. In more than 30 years of research on multivariate cryptography, results on constructing signature schemes seem to be more fruitful, UOV [KPG99] and Rainbow [DS05] remains secure after many years of attack attempts. On the other hand, history on multivariate encryption is more turbulent. Many multivariate encryption schemes have been proposed, such as MI [MI88], HFE [Pat96], ABC [TDTD13], ZHFE [PBD14], SRP [YS16], EFC [SDP16], HFERP [IPST⁺18] and EFLASH [CST18]. Most of them were proven to be insecure under various attacks, such as MinRank [GC00b], HighRank [CSV93a], Linearization [Pat95]. Nevertheless, ABC, EFC, HFERP and EFLASH still remain secure. At PQCrypto 2016, Szepieniec et al. [SDP16] proposed multivariate encryption schemes EFC_p^- and $\text{EFC}_{pt^2}^-$, which use matrix multiplications as in ABC [TDTD13], and extension field structure as in MI [MI88], HFE [Pat96] and ZHFE [PBD14].

In this chapter, we introduce a more efficient decryption approach for EFC_p^- and $\text{EFC}_{pt^2}^-$. The decryption algorithms for EFC_p^- and $\text{EFC}_{pt^2}^-$ rely on the bilinear relation between the plaintext and an augmented ciphertext, that is the concatenation of a ciphertext and the values of the deleted polynomials by the minus modifier. This bilinear relation is used for constructing linear systems in the decryption process of EFC_p^- and $\text{EFC}_{pt^2}^-$. Our proposed decryption algorithms aim to separate the computation of con-

structuring the linear system into two kinds of computations. One is the computation involving the plaintext and the ciphertext. The other one is computation involving the plaintext and the guessed values. In addition, we experimentally investigate the security of EFC_p^- and $\text{EFC}_{pt^2}^-$ through hybrid attack [BFP09], which is a combination of algebraic attack and exhaustive search.

This chapter is structured as follows. In Section 4.2, we recall multivariate cryptography. In Section 4.3, we recall the construction of EFC_p^- and $\text{EFC}_{pt^2}^-$, and their decryption algorithms in [SDP16]. In Section 4.4, we introduce our proposed new decryption algorithms for EFC_p^- and $\text{EFC}_{pt^2}^-$. In Section 4.5, we apply hybrid algebraic attack on EFC_p^- and $\text{EFC}_{pt^2}^-$ and estimate new secure parameters for them. Finally, we conclude the chapter in Section 4.6.

4.2 Multivariate cryptography

In this section, we give a short introduction to multivariate cryptography. Let \mathbb{F} denote a finite field with q elements, n, m be two positive integers and denote the polynomial ring in variables x_1, \dots, x_n over \mathbb{F} by $\mathbb{F}[x_1, \dots, x_n]$.

4.2.1 Quadratic maps and MQ problem

In this subsection, we introduce the notion of quadratic maps and MQ-problem.

Given quadratic polynomials $f_1, \dots, f_m \in \mathbb{F}[x_1, \dots, x_n]$,

$$f_k = \sum_{i=1}^n \sum_{j=1}^n a_{ij} x_i x_j + \sum_{i=1}^n b_i x_i + c, \quad (1 \leq k \leq m)$$

where $a_{ij}, b_i, c \in \mathbb{F}$, then $F(x_1, \dots, x_n) = (f_1, \dots, f_m) : \mathbb{F}^n \rightarrow \mathbb{F}^m$ is called a *quadratic map*.

Multivariate cryptography refers to the study of public key cryptosystems whose public keys are quadratic maps. The security of multivariate cryptography is based on *MQ problem*, that is, given m quadratic polynomials $p_1, \dots, p_m \in \mathbb{F}[x_1, \dots, x_n]$, find $\mathbf{z} \in \mathbb{F}^n$ such that $p_1(\mathbf{z}) = \dots = p_m(\mathbf{z}) = 0$. MQ problem is proven to be NP-complete even for the simplest case of multivariate quadratic polynomials over $\text{GF}(2)$. Therefore, multivariate cryptography is considered to be a candidate for post-quantum cryptography.

4.2.2 Construction of multivariate encryption schemes

To construct a multivariate encryption scheme, we need to design its private key, public key, encryption and decryption processes.

Private key We start with choosing an easy-to-invert quadratic map $F(x_1, \dots, x_n) = (f_1, \dots, f_m)$. “Easy-to-invert” means given the value $(y_1, \dots, y_m) \in \mathbb{F}^m$, solving the system $f_i = y_i$ ($1 \leq i \leq m$) is easy. Such map F is also called a *central map*. Then we choose two invertible linear maps $S : \mathbb{F}^n \rightarrow \mathbb{F}^n$ and $T : \mathbb{F}^m \rightarrow \mathbb{F}^m$. The set $\{F, S, T\}$ is a private key.

Public key Given a private key $\{F, S, T\}$, a public key P can be generated from the composition of F, S, T , i.e. $P = T \circ F \circ S$.

General workflow Given a public key P and a plaintext $\mathbf{z} \in \mathbb{F}^n$, the ciphertext $\mathbf{c} \in \mathbb{F}^m$ of \mathbf{z} can be obtained by performing $\mathbf{c} = P(\mathbf{z})$.

Conversely, given a private key $\{F, S, T\}$ and a ciphertext $\mathbf{c} \in \mathbb{F}^m$, a multivariate encryption scheme decrypts \mathbf{c} by computing the inverse of T, F and S individually.

4.2.3 Algebraic attack

Algebraic attack directly solves the system:

$$\begin{aligned} p_1(x_1, \dots, x_n) &= c_1, \\ &\vdots \\ p_m(x_1, \dots, x_n) &= c_m, \end{aligned} \tag{4.1}$$

where p_i ’s are public key polynomials, and $\mathbf{c} = (c_1, \dots, c_m)$ is a ciphertext. Usually, Gröbner basis method is used in the solving process.

Gröbner basis method was introduced by Buchberger, who proposed Buchberger algorithm [Buc65], and was later improved by Faugere with F4/F5 algorithms, see [Fau99, Fau02]. “Field equations”

$$\begin{aligned} x_1^q - x_1 &= 0, \\ &\vdots \\ x_n^q - x_n &= 0, \end{aligned}$$

are added to the polynomial system (equation (4.1)) to solve this system in \mathbb{F} , and it is much faster to compute a Gröbner basis when field equations are added. In multivariate cryptography, let

$$G = \{p_1 - c_1, \dots, p_m - c_m, x_1^q - x_1, \dots, x_n^q - x_n\}, \quad (4.2)$$

we then need to compute the Gröbner basis of the ideal \mathcal{I} generated by G . The complexity of the Gröbner basis method is determined by a so-called *degree of regularity*. There are several different definitions for degree of regularity, through out this paper, we regard the degree of regularity as the degree where a non-trivial syzygy producing a degree fall first occurs, which is also called the *first fall degree* [JHPS14]. If F4 or F5 algorithm is used, the complexity of computing the Gröbner basis of \mathcal{I} is

$$\text{Complexity}_{F4/F5} = O\left(\binom{n + d_{reg} - 1}{d_{reg}}^2 \binom{n}{2}\right), \quad (4.3)$$

where d_{reg} is the degree of regularity of \mathcal{I} .

In [BFP09], Bettale et al. proposed a hybrid algebraic method for solving multivariate systems over finite fields, which can speed up the computation of Gröbner basis with F4/F5 algorithms. Hybrid algebraic method is a combination of algebraic attack and exhaustive search. Specifically, given $e \in \mathbb{N}$, hybrid algebraic method first guesses the value of variables x_1, \dots, x_e , evaluate them in G with the guessing values, and then apply algebraic attack on the remaining polynomial system. The guessing processes terminates when a solution is found.

4.3 Extension field cancellation (EFC)

In this section, we recall the constructions of EFC_p^- and $\text{EFC}_{pt^2}^-$ [SDP16], and the original decryption algorithms designed for them.

4.3.1 Notations

Let \mathbb{F} be a finite field of 2 elements. Given a positive integer n , x_1, \dots, x_n are n variables over \mathbb{F} , and define $\mathbf{x} = (x_1, \dots, x_n)$. \mathbb{E} denotes a degree n extension field of \mathbb{F} . Denote the set of all $n \times m$ matrices by $\mathbb{F}^{n \times m}$. Matrices are denoted by capital letters, vectors are denoted by bold lowercase letters, and all vectors are treated as row vectors.

The i -th entry of a vector \mathbf{v} is denoted by v_i , the i -th row of a matrix M is denoted by M_i . For $N, M \in \mathbb{F}^{n \times n}$, $(N || M) \in \mathbb{F}^{n \times 2n}$ denotes the horizontal join of N and M .

Choose a basis $\{\theta_1, \dots, \theta_n\}$ of \mathbb{E}/\mathbb{F} , and define the isomorphism

$$\begin{aligned} \varphi : \mathbb{F}^n &\rightarrow \mathbb{E} \\ \mathbf{v} &\mapsto \mathbf{v}\mathbf{b}^\top, \end{aligned}$$

where $\mathbf{b} = (\theta_1, \dots, \theta_n) \in \mathbb{E}^n$. For $A \in \mathbb{F}^{n \times n}$, and $\mathbf{v} = (v_1, \dots, v_n) \in \mathbb{F}^n$, define $\alpha(\mathbf{v}) = \varphi(\mathbf{v}A) \in \mathbb{E}$. The matrix associated with the linear map

$$\begin{aligned} \mathbb{E} &\rightarrow \mathbb{E} \\ X &\mapsto \alpha(\mathbf{v})X \end{aligned}$$

is denoted by $\alpha_m(\mathbf{v}) \in \mathbb{F}^{n \times n}$. For a matrix $B \in \mathbb{F}^{n \times n}$ and $\mathbf{v} \in \mathbb{F}^n$, we define $\beta(\mathbf{v})$ and $\beta_m(\mathbf{v})$ in the same way as $\alpha(\mathbf{v})$ and $\alpha_m(\mathbf{v})$. For a positive integer a , π_a stands for the following projection:

$$\begin{aligned} \pi_a : \mathbb{F}^{2n} &\rightarrow \mathbb{F}^{2n-a} \\ (v_1, \dots, v_{2n}) &\mapsto (v_1, \dots, v_{2n-a}). \end{aligned}$$

4.3.2 Construction of the EFC_p^- scheme

Key generation Given a prime number n , randomly choose $A, B \in \mathbb{F}^{n \times n}$ of rank $n-1$ such that the intersection of the kernel spaces of A and B is the zero subspace. Randomly choose two invertible linear maps $S : \mathbb{F}^n \rightarrow \mathbb{F}^n$ and $T : \mathbb{F}^{2n} \rightarrow \mathbb{F}^{2n}$, we identify these linear maps with matrices $S \in \mathbb{F}^{n \times n}$, $T \in \mathbb{F}^{2n \times 2n}$. The central map F for EFC_p^- is

$$\begin{aligned} F : \mathbb{F}^n &\rightarrow \mathbb{F}^{2n} \\ \mathbf{x} &\mapsto (\mathbf{x} \cdot \alpha_m(\mathbf{x}), \mathbf{x} \cdot \beta_m(\mathbf{x})). \end{aligned}$$

The public key for EFC_p^- is given by

$$P = (p_1, \dots, p_{2n-a}) = \pi_a \circ T \circ F \circ S : \mathbb{F}^n \rightarrow \mathbb{F}^{2n-a},$$

where p_i ($1 \leq i \leq 2n-a$) are quadratic polynomials in x_1, \dots, x_n over \mathbb{F} .

Next we take a look at the explicit form of the central map F . For any $\mathbf{x} \in \mathbb{F}^n$, $\alpha(\mathbf{x}) \in \mathbb{E}$ can be represented with basis $\{\theta_1, \dots, \theta_n\}$, i.e. $\alpha(\mathbf{x}) = \mathbf{x}A\mathbf{b}^\top$. Let

$\alpha_i = A_i \mathbf{b}^\top \in \mathbb{E}$ for $1 \leq i \leq n$, then we have $\alpha(\mathbf{x}) = \sum_{i=1}^n x_i \alpha_i$. Define matrices $C^{(i)} \in \mathbb{F}^{n \times n}$ by $(C^{(i)})_j^\top = \varphi^{-1}(\alpha_i \theta_j)$ for $1 \leq i, j \leq n$. It is easy to check that $C^{(i)}$ satisfies $\mathbf{b} C^{(i)} = \alpha_i \mathbf{b}$ for $1 \leq i \leq n$, which indicates $\alpha_m(\mathbf{x}) = \sum_{i=1}^n x_i C^{(i)}$. Similarly, we define matrices $D^{(i)} \in \mathbb{F}^{n \times n}$ for $1 \leq i \leq n$ and they satisfy $\beta_m(\mathbf{x}) = \sum_{i=1}^n x_i D^{(i)}$. Therefore, the explicit form of F is

$$F : \mathbb{F}^n \rightarrow \mathbb{F}^{2n}$$

$$\mathbf{x} \mapsto \left(\mathbf{x} \cdot \left(\sum_{i=1}^n C^{(i)} x_i \right), \mathbf{x} \cdot \left(\sum_{i=1}^n D^{(i)} x_i \right) \right).$$

Encryption Given a public key P and a plaintext $\mathbf{z} \in \mathbb{F}^n$, its ciphertext is $\mathbf{c} = P(\mathbf{z}) \in \mathbb{F}^{2n-a}$.

Decryption Given the private key $\{A, B, S, T\}$ and a ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$, decryption process is to find the plaintext $\mathbf{z} \in \mathbb{F}^n$ such that $P(\mathbf{z}) = \mathbf{c}$. First, we need to guess the value \mathbf{v} from \mathbb{F}^a for the deleted polynomials by π_a . Second, we compute $\mathbb{F}^n \times \mathbb{F}^n \ni (\mathbf{d}_1, \mathbf{d}_2) = \mathbf{d} = T^{-1}(\mathbf{c}, \mathbf{v})$. Next we invert the map F by solving the linear system

$$\mathbf{d}_2 \alpha_m(\mathbf{x}) = \mathbf{d}_1 \beta_m(\mathbf{x}), \quad (4.4)$$

and obtain a solution $\mathbf{h} \in \mathbb{F}^n$. Finally, if $F(\mathbf{h}) = (\mathbf{d}_1, \mathbf{d}_2)$, then we obtain the plaintext by $\mathbf{z} = S^{-1}(\mathbf{h})$. The loop of guessing the value \mathbf{v} from \mathbb{F}^a terminates when the correct plaintext \mathbf{z} is found. The details are shown in Algorithm 4.1.

Regrading the complexity of this decryption algorithm, we have the following proposition:

Proposition 4.3.1. The number of field operations involved in the decryption algorithm for EFC_p^- is

$$8n^4 + 9n^3 + \frac{1}{2}n^2 - \frac{7}{2}n + 2^{(a-1)} \left(\frac{26}{3}n^3 + \frac{21}{2}n^2 - \frac{31}{6}n \right). \quad (4.5)$$

Proof. Let $[+]_{\mathbb{F}}$ denotes \mathbb{F} -addition, and $[\times]_{\mathbb{F}}$ denotes \mathbb{F} -multiplication of \mathbb{F} . We recall the complexity of Gaussian Elimination, and multiplication in \mathbb{E} . For an input of $n \times m$ ($m \geq n$) matrix over \mathbb{F} , Gaussian Elimination requires $\sum_{i=1}^{n-1} (n-i)(m-i) [+]_{\mathbb{F}}$ and $\sum_{i=1}^{n-1} (n-i)(m-i) + \sum_{i=1}^{n-1} (n-i) [\times]_{\mathbb{F}}$. For any $a, b \in \mathbb{E}$, that are

Algorithm 4.1: Decryption algorithm for EFC_p⁻ [SDP16]**Input** : A ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$,The private key $A, B, S \in \mathbb{F}^{n \times n}$ and $T \in \mathbb{F}^{2n \times 2n}$.**Output:** The plaintext $\mathbf{z} \in \mathbb{F}^n$.

```

1  $S_{inv} \leftarrow S^{-1}, T_{inv} \leftarrow T^{-1}$ 
2 Generate  $\alpha_m(\mathbf{x}), \beta_m(\mathbf{x})$  and  $F$  from  $A, B$ 
3 for  $\mathbf{v} \in \mathbb{F}^a$  do
4    $\mathbb{F}^n \times \mathbb{F}^n \ni (\mathbf{d}_1, \mathbf{d}_2) = \mathbf{d} \leftarrow (\mathbf{c}, \mathbf{v}) \cdot T_{inv}$ 
5   construct a linear system  $\mathbf{d}_2 \cdot \alpha_m(\mathbf{x}) - \mathbf{d}_2 \cdot \beta_m(\mathbf{x}) = 0$ 
6    $\mathbf{x} = \mathbf{h} \leftarrow \text{solve } \mathbf{d}_2 \cdot \alpha_m(\mathbf{x}) - \mathbf{d}_2 \cdot \beta_m(\mathbf{x}) = 0$ 
7   if  $F(\mathbf{h}) = \mathbf{d}$  then
8      $\text{break}$ 
9  $\mathbb{F}^n \ni \mathbf{z} \leftarrow \mathbf{h} \cdot S_{inv}$ 
10 Return  $\mathbf{z}$ .
```

represented in basis $\{\theta_1, \dots, \theta_n\}$, the multiplication $a \cdot b$ requires $(n-1)(2n-1)$ $[+]_{\mathbb{F}}$ and $2n^2$ $[\times]_{\mathbb{F}}$.

Now we analyze the complexity based on the Algorithm 4.1.

In step 1, computing T^{-1} requires $\frac{n(20n^2-12n+1)}{3}$ $[+]_{\mathbb{F}}$ and $\frac{2n(10n^2-3n-1)}{3}$ $[\times]_{\mathbb{F}}$, and computing S^{-1} requires $\frac{n(5n^2-6n+1)}{6}$ $[+]_{\mathbb{F}}$ and $\frac{n(5n^2-3n-2)}{6}$ $[\times]_{\mathbb{F}}$. In step 2, to obtain $\alpha_m(\mathbf{x})$, we need to compute $\alpha(\mathbf{x}) = \sum_{i=1}^n x_i \alpha_i$, where $\alpha_i = A_i \mathbf{b}^\top$ ($1 \leq i \leq n$), and this requires $n(n-1)$ $[+]_{\mathbb{F}}$ and n^2 $[\times]_{\mathbb{F}}$. Then we need to compute $\alpha_i \mathbf{b}$ for $1 \leq i \leq n$, which indicates n^2 $[\times]_{\mathbb{F}}$, and it requires $n^2(n-1)(2n-1)$ $[+]_{\mathbb{F}}$ and $2n^4$ $[\times]_{\mathbb{F}}$. Same complexity holds for obtaining $\beta_m(\mathbf{x})$.

From step 3 to step 8, we enter a loop of size 2^a . In step 4, $(\mathbf{c}, \mathbf{v}) \cdot T_{inv}$ requires $2n(2n-1)$ $[+]_{\mathbb{F}}$ and $4n^2$ $[\times]_{\mathbb{F}}$. In step 5, constructing the linear system needs $2n^3 - n^2$ $[+]_{\mathbb{F}}$ and $2n^3$ $[\times]_{\mathbb{F}}$. In step 6, solving the linear system with Gaussian Elimination requires $\frac{n(n-1)(2n+5)}{6}$ $[+]_{\mathbb{F}}$ and $\frac{n(n^2+3n-1)}{3}$ $[\times]_{\mathbb{F}}$. In step 7, verifying whether $F(\mathbf{h}) = \mathbf{d}$ holds costs $2n(n^2-1)$ $[+]_{\mathbb{F}}$ and $2n^2(n+1)$ $[\times]_{\mathbb{F}}$. The loop terminates in step 8 after an average of $2^{(a-1)}$ times. Therefore, the loop costs $2^{(a-1)}(\frac{13}{3}n^3 + \frac{7}{2}n^2 - \frac{29}{6}n)$ $[+]_{\mathbb{F}}$ and $2^{(a-1)}(\frac{13}{3}n^3 + 7n^2 - \frac{1}{3}n)$ $[\times]_{\mathbb{F}}$ in average.

In step 9, computing $\mathbf{h} \cdot S_{inv}$ needs $n(n-1)$ $[+]_{\mathbb{F}}$ and n^2 $[\times]_{\mathbb{F}}$.

Since step 1, step 2 and step 9 together costs $4n^4 + \frac{3}{2}n^3 - \frac{5}{2}n$ $[+]_{\mathbb{F}}$ and $4n^4 + \frac{15}{2}n^3 + \frac{1}{2}n^2 - n$ $[\times]_{\mathbb{F}}$, the total cost of this decryption algorithm is Equation (4.5). This completes the proof. \square

4.3.3 Construction of the $\text{EFC}_{pt^2}^-$ scheme

Key generation Choose the secret key A, B and S, T as in EFC_p^- . The central map F for $\text{EFC}_{pt^2}^-$ is

$$F : \mathbb{F}^n \rightarrow \mathbb{F}^{2n} \\ \mathbf{x} \mapsto (\mathbf{x}\alpha_m(\mathbf{x}) + \varphi^{-1}(\beta(\mathbf{x})^3), \mathbf{x}\beta_m(\mathbf{x}) + \varphi^{-1}(\alpha(\mathbf{x})^3)). \quad (4.6)$$

The public key for $\text{EFC}_{pt^2}^-$ is $P = (p_1, \dots, p_{2n-a}) = \pi_a \circ T \circ F \circ S : \mathbb{F}^n \rightarrow \mathbb{F}^{2n-a}$. The private key consists of A, B and S, T .

We take a look at the explicit structure of (4.6) using $\mathbf{b} = (\theta_1, \dots, \theta_n)$. Since $\mathbf{x} \cdot \alpha_m(\mathbf{x})$ and $\mathbf{x} \cdot \beta_m(\mathbf{x})$ can be represented in the same way as in Section 4.3.2, we show the explicit form of $\varphi^{-1}(\alpha(\mathbf{x}))$ and $\varphi^{-1}(\beta(\mathbf{x}))$ here. Let $\Theta = \mathbf{b}^\top \cdot \mathbf{b} \in \mathbb{E}^{n \times n}$ and $\varphi^{-1}(\Theta) = (\Theta_1, \dots, \Theta_n) \in (\mathbb{F}^{n \times n})^n$. Define a matrix $\Delta \in \mathbb{F}^{n \times n}$ by its i -th row $\Delta_i = \varphi^{-1}(\theta_i^2)$. Then $\alpha(\mathbf{x})^3$ can be represented as

$$\begin{aligned} \alpha(\mathbf{x})^3 &= \alpha(\mathbf{x})^2 \cdot \alpha(\mathbf{x}) = \mathbf{x}A (\theta_1^2, \dots, \theta_n^2)^\top \cdot \mathbf{b}(\mathbf{x}A)^\top \\ &= \mathbf{x}A\Delta\Theta(\mathbf{x}A)^\top = \sum_{i=1}^n \theta_i \cdot \mathbf{x}A\Delta\Theta_i(\mathbf{x}A)^\top. \end{aligned}$$

$\beta(\mathbf{x})^3$ can be represented in the same way. Therefore,

$$\begin{aligned} \varphi^{-1}(\alpha(\mathbf{x})^3) &= (\mathbf{x}A\Delta\Theta_1(\mathbf{x}A)^\top, \dots, \mathbf{x}A\Delta\Theta_n(\mathbf{x}A)^\top), \\ \varphi^{-1}(\beta(\mathbf{x})^3) &= (\mathbf{x}B\Delta\Theta_1(\mathbf{x}B)^\top, \dots, \mathbf{x}B\Delta\Theta_n(\mathbf{x}B)^\top). \end{aligned}$$

Encryption Given a public key P and a plaintext $\mathbf{z} \in \mathbb{F}^n$, the ciphertext is $\mathbf{c} = P(\mathbf{z}) \in \mathbb{F}^{2n-a}$.

Decryption We take a look at how to invert the central map F . It requires solving the system $F(\mathbf{x}) = \mathbf{d} \in \mathbb{F}^{2n}$, i.e.

$$\mathbf{x} \cdot \alpha_m(\mathbf{x}) + \varphi^{-1}(\beta(\mathbf{x})^3) = \mathbf{d}_1, \quad \mathbf{x} \cdot \beta_m(\mathbf{x}) + \varphi^{-1}(\alpha(\mathbf{x})^3) = \mathbf{d}_2, \quad (4.7)$$

where $\mathbf{d} = (\mathbf{d}_1, \mathbf{d}_2) \in \mathbb{F}^n \times \mathbb{F}^n$. By definition of $\alpha_m(\mathbf{x})$ in Section 4.3.1, the equation $\varphi(\mathbf{x} \cdot \alpha_m(\mathbf{x})) = \varphi(\mathbf{x})\alpha(\mathbf{x})$ holds. Thus (4.7) is equivalent to

$$\varphi(\mathbf{x})\alpha(\mathbf{x}) + \beta(\mathbf{x})^3 = \varphi(\mathbf{d}_1), \quad \varphi(\mathbf{x})\beta(\mathbf{x}) + \alpha(\mathbf{x})^3 = \varphi(\mathbf{d}_2),$$

from which the following system can be constructed:

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \varphi^{-1}(\alpha(\mathbf{x})^4 - \beta(\mathbf{x})^4). \quad (4.8)$$

Define a matrix $\Lambda \in \mathbb{F}^{n \times n}$ by $\Lambda_i = \varphi^{-1}(\theta_i^4)$ for $1 \leq i \leq n$, and apply it to (4.8). Then (4.8) turns into

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \mathbf{x}(A - B)\Lambda, \quad (4.9)$$

which is a linear system in \mathbf{x} . The rest of the procedures of decryption is similar to that of EFC_p^- , details are shown in Algorithm 4.2.

Algorithm 4.2: Decryption algorithm for $\text{EFC}_{pt^2}^-$ [SDP16]

Input : $\mathbf{b} = (\theta_1, \dots, \theta_n) \in \mathbb{E}^n$, a ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$,
the private key $A, B, S \in \mathbb{F}^{n \times n}$ and $T \in \mathbb{F}^{2n \times 2n}$.

Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$.

```

1  $S_{inv} \leftarrow S^{-1}, T_{inv} \leftarrow T^{-1}$ 
2 Define  $\Lambda \in \mathbb{F}^{n \times n}$  by  $\Lambda_i = \varphi^{-1}(\theta_i^4)$ 
3 Generate  $\alpha_m(\mathbf{x}), \beta_m(\mathbf{x})$  and  $F$  from  $A, B$ 
4 for  $\mathbf{v} \in \mathbb{F}^a$  do
5    $\mathbb{F}^n \times \mathbb{F}^n \ni (\mathbf{d}_1, \mathbf{d}_2) = \mathbf{d} \leftarrow (\mathbf{c}, \mathbf{v}) \cdot T_{inv}$ 
6   construct a linear system  $\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \mathbf{x}(A - B)\Lambda$ 
7    $\mathbf{x} = \mathbf{h} \leftarrow$  solve  $\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \mathbf{x}(A - B)\Lambda$ 
8   if  $F(\mathbf{h}) = \mathbf{d}$  then
9     break
10  $\mathbb{F}^n \ni \mathbf{z} \leftarrow \mathbf{h} \cdot S_{inv}$ 
11 Return  $\mathbf{z}$ .
```

We analyze the complexity of the decryption algorithm for $\text{EFC}_{pt^2}^-$ adopting the same approach as in the proof of Proposition 4.3.1, and obtain the number of field

operations involved in the decryption algorithm for $\text{EFC}_{pt^2}^-$ as

$$8n^4 + 17n^3 - \frac{11}{2}n^2 - \frac{3}{2}n + 2^{(a-1)} \left(\frac{32}{3}n^3 + \frac{19}{2}n^2 - \frac{31}{6}n \right). \quad (4.10)$$

4.4 Proposed efficient decryption algorithms for EFC

In this section, we introduce our new decryption algorithms for EFC_p^- and $\text{EFC}_{pt^2}^-$.

4.4.1 New decryption algorithm for EFC_p^-

The new decryption algorithm is derived from linearization equations, which represent a relation between the plaintext and ciphertext.

We begin with deriving linearization equations related to the central map of EFC_p^- . Recall that the linear system (4.4) for inverting its central map is

$$\mathbf{d}_2 \alpha_m(\mathbf{x}) - \mathbf{d}_1 \beta_m(\mathbf{x}) = 0,$$

which can also be written as

$$\alpha(\mathbf{x})\varphi(\mathbf{d}_2) - \beta(\mathbf{x})\varphi(\mathbf{d}_1) = \mathbf{x}A\mathbf{b}^\top \cdot \mathbf{b}\mathbf{d}_2^\top - \mathbf{x}B\mathbf{b}^\top \cdot \mathbf{b}\mathbf{d}_1^\top = 0.$$

Let $\Theta = \mathbf{b}^\top \cdot \mathbf{b}$ and $(\Theta_1, \dots, \Theta_n) = \varphi^{-1}(\Theta)$, then from this equation, we can obtain linearization equations corresponding to the central map of EFC_p^- as follows:

$$\mathbf{x}(B\Theta_i || A\Theta_i)\mathbf{d}^\top = 0, \quad (1 \leq i \leq n), \quad (4.11)$$

where $\mathbf{d} = (\mathbf{d}_1, \mathbf{d}_2)$.

Subsequently, we apply S and T to the linearization equation related to the central map. Let $\mathbf{c} \in \mathbb{F}^{2n}$ be a ciphertext of EFC_p^- without minus modifier, then $\mathbf{c} = T(\mathbf{d})$. Apply the linear maps S and T to Equation (4.11), and we obtain the linearization equations between a plaintext \mathbf{x} and \mathbf{c} as

$$\mathbf{x}S(B\Theta_i || A\Theta_i)(\mathbf{c}T^{-1})^\top = 0. \quad (4.12)$$

For a ciphertext \mathbf{c} of EFC_p^- without minus modifier, its corresponding plaintext can be found by solving Equation (4.12).

Next we show how to represent Equation (4.12) into one simple equation. Let $N^{(i)} = T^{-1}(SB\Theta_i || SA\Theta_i)^\top \in \mathbb{F}^{2n \times n}$, and define matrices $U^{(j)}$ by $U_i^{(j)} = N_j^{(i)}$ for $1 \leq j \leq 2n$ and $1 \leq i \leq n$. Then Eq (4.12) turns into one simple equation

$$(c_1 U^{(1)} + \dots + c_{2n} U^{(2n)}) \cdot \mathbf{x}^\top = 0. \quad (4.13)$$

This equation indicates that as long as we have the set $\Psi = (U^{(1)}, \dots, U^{(2n)})$, the decryption process of EFC_p^- without minus modifier can be reduced into the computation of the right kernel space of $c_1 U^{(1)} + \dots + c_{2n} U^{(2n)}$.

Since in our new decryption algorithm, only the ciphertext \mathbf{c} and $U^{(1)}, \dots, U^{(2n)}$ are necessary, we intend to save $\Psi = (U^{(1)}, \dots, U^{(2n)})$ as the new private key for EFC_p^- , which is $2n/7$ times larger than the original private key.

Now we explain our proposed decryption algorithm for EFC_p^- . First, we compute $L = \sum_{i=1}^{2n-a} c_i U^{(i)}$. Second, we guess the values for the deleted polynomials by π_a from \mathbb{F}^a , and denote these values by $\mathbf{v} = (v_1, \dots, v_a)$. Next, we compute the right kernel space $\mathbf{ker} = \ker(L + \sum_{i=1}^a v_i U^{(2n-a+i)})$. Finally, we check if there exists $\mathbf{z} \in \mathbf{ker}$ such that $P(\mathbf{z}) = \mathbf{c}$ holds. If so, then \mathbf{z} is the plaintext, otherwise, go back to the guessing step and start over. The details of the procedures of generating Ψ and the decryption process are shown in Algorithm 4.3.

Remark 4.4.1. The iterative computation complexity of $\sum_{i=1}^a v_i U^{(2n-a+i)}$ can be further reduced. Assume we have $\mathbf{v}^{(1)}, \mathbf{v}^{(2)} \in \mathbb{F}_2^a$, and we need to compute

$$L^{(1)} = \sum_{i=1}^a v_i^{(1)} U^{(2n-a+i)}, \quad L^{(2)} = \sum_{i=1}^a v_i^{(2)} U^{(2n-a+i)}.$$

If we know the difference between $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$, we will be able to compute $L^{(2)}$ from $L^{(1)}$ by subtracting a few matrices. Since Gray code has the property that two successive binary numeral values differ in only one bit, we use Gray code to represent \mathbb{F}_2^a . Assume $\mathbf{v}^{(1)}$ and $\mathbf{v}^{(2)}$ are two successive codes, and they differ at j -th bit, then we can compute $L^{(2)}$ by

$$L^{(2)} = L^{(1)} + U^{(2n-a+j)}.$$

Therefore, this technique reduces the number of operations of matrix addition involved in recursive computation of $\sum_{i=1}^a v_i U^{(2n-a+i)}$. We consider this technique when we evaluate the complexity of our new decryption.

Regarding the complexity of the new decryption algorithm for EFC_p^- , we have the following proposition.

Proposition 4.4.1. The number of field operations involved in the new decryption for

Algorithm 4.3: New decryption algorithm for EFC_p^-

Input : $\mathbf{b} = (\theta_1, \dots, \theta_n)$, the private key $A, B, S \in \mathbb{F}^{n \times n}$ and $T \in \mathbb{F}^{2n \times 2n}$, a ciphertext $\mathbf{c} \in \mathbb{F}^{2n-a}$.

Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$ s.t. $P(\mathbf{z}) = \mathbf{c}$.

```

1  $\Theta \leftarrow \mathbf{b}^\top \cdot \mathbf{b}$ ,  $(\Theta_1, \dots, \Theta_n) \leftarrow \varphi^{-1}(\Theta)$ 
2 for  $i \leftarrow 1$  to  $n$  do
3    $N^{(i)} \leftarrow T^{-1}(SB\Theta_i || SA\Theta_i)^\top \in \mathbb{F}^{2n \times n}$ 
4 for  $j \leftarrow 1$  to  $2n$  and  $i \leftarrow 1$  to  $n$  do
5    $U_i^{(j)} \leftarrow N_j^{(i)}$ 
6  $L \leftarrow \sum_{i=1}^{2n-a} c_i U^{(i)}$ 
7 for  $\mathbf{v} = (v_1, \dots, v_a) \in \mathbb{F}^a$  do
8    $H \leftarrow L + \sum_{i=1}^a v_i U^{(2n-a+i)}$ 
9    $\text{ker} \leftarrow \text{RightKer}(H)$ 
10  for  $\mathbf{z} \in \text{ker}$  do
11    if  $P(\mathbf{z}) = \mathbf{c}$  then
12      Return  $\mathbf{z}$ 
13    break
```

EFC_p^- is

$$2^{(a-1)} \left(\frac{14}{3}n^3 + \left(\frac{11}{2} - 2a \right)n^2 - \left(a + \frac{19}{6} \right)n + a \right) + 4n^3 - (2a+1)n^2 \quad (4.14)$$

Proof. Let $[+]_{\mathbb{F}}$ denote \mathbb{F} -addition, and $[\times]_{\mathbb{F}}$ denote the \mathbb{F} -multiplication. We analyze the complexity based on Algorithm 4.3. Note that we analyze the complexity starting from step 6 since we save Ψ as the new private key.

In step 6, $\sum_{i=1}^{2n-a} c_i U^{(i)}$ requires $n^2(2n-a-1) [+]_{\mathbb{F}}$ and $n^2(2n-a) [\times]_{\mathbb{F}}$.

From step 7 to 13, we enter a loop of size 2^a . In step 8, $L + \sum_{i=1}^a v_i U^{(2n-a+i)}$ costs $2n^2 [+]_{\mathbb{F}}$ using Gray code technique in Remark 1. In step 9, finding the right kernel of H requires $\frac{n(n-1)(2n+5)}{6} [+]_{\mathbb{F}}$ and $\frac{n(n^2+3n-1)}{3} [\times]_{\mathbb{F}}$. In step 11, verifying the solution requires $(2n-a)(n^2-1) [+]_{\mathbb{F}}$ and $(2n-a)(n^2+n) [\times]_{\mathbb{F}}$. In step 13, the loop terminates after an average of $2^{(a-1)}$ times. Therefore, the loop requires $2^{(a-1)} \left(\frac{7}{3}n^3 + \left(\frac{5}{2} - a \right)n^2 - \frac{17}{6}n + a \right)$

$[+]_{\mathbb{F}}$ and $2^{(a-1)}(\frac{7}{3}n^3 + (3-a)n^2 - (a + \frac{1}{3})n) [\times]_{\mathbb{F}}$ in average.

Therefore, the total cost of this decryption algorithm is Equation (4.14). This completes the proof. \square

4.4.2 New decryption algorithm for $\text{EFC}_{pt^2}^-$

Same as EFC_p^- , the new decryption algorithm for $\text{EFC}_{pt^2}^-$ also derives from linearization equations.

We first consider linearization equations related to the central map of $\text{EFC}_{pt^2}^-$. Recall in Section 4.3.3, inverting the central map of $\text{EFC}_{pt^2}^-$ requires solving the linear system

$$\mathbf{d}_2\alpha_m(\mathbf{x}) - \mathbf{d}_1\beta_m(\mathbf{x}) = \mathbf{x}(A - B)\Lambda, \quad (4.15)$$

where $\Lambda \in \mathbb{F}^{n \times n}$, $\Lambda_i = \varphi^{-1}(\theta_i^4)$ for $1 \leq i \leq n$. This equation can also be written as

$$\varphi(\mathbf{d}_2)\alpha(\mathbf{x}) - \varphi(\mathbf{d}_1)\beta(\mathbf{x}) = \varphi(\mathbf{x}(A - B)\Lambda). \quad (4.16)$$

Applying φ^{-1} on both sides of Equation (4.16) gives us

$$\mathbf{x}(B\Theta_i || A\Theta_i)\mathbf{d}^\top - (\mathbf{x}(A - B)\Lambda)_i = 0, \quad (1 \leq i \leq n), \quad (4.17)$$

which are the linearization equations related to the central map of $\text{EFC}_{pt^2}^-$.

Next we apply S and T to the linearization equations we obtained. Let $\mathbf{c} \in \mathbb{F}^{2n}$ be a ciphertext of $\text{EFC}_{pt^2}^-$ without minus modifier, then $\mathbf{c} = T(\mathbf{d})$. Applying linear maps S and T on (4.17) gives us the linearization equations of a plaintext \mathbf{x} and a ciphertext \mathbf{c} :

$$\mathbf{x}(SB\Theta_i || SA\Theta_i)(\mathbf{c}T^{-1})^\top - (\mathbf{x}S(A - B)\Lambda)_i = 0. \quad (4.18)$$

Next we show how to represent Equation (4.18) into one simple equation. Let $M = S(A - B)\Lambda \in \mathbb{F}^{n \times n}$, $N^{(i)} = T^{-1}(SB\Theta_i || SA\Theta_i)^\top \in \mathbb{F}^{2n \times n}$, and define matrices $U^{(j)}$ by $U_i^{(j)} = N_j^{(i)}$ for $1 \leq j \leq 2n$ and $1 \leq i \leq n$. Then (4.18) can be rearranged into

$$(c_1U^{(1)} + \dots + c_{2n}U^{(2n)} - M^\top) \cdot \mathbf{x}^\top = 0. \quad (4.19)$$

This equation indicates that the decryption of $\text{EFC}_{pt^2}^-$ without minus modifier can be reduced to the computation of the right kernel space of $c_1U^{(1)} + \dots + c_{2n}U^{(2n)} - M^\top$.

Similar to EFC_p^- , we save $\Psi = (U^{(1)}, \dots, U^{(2n)}, M)$ as the new private key for $\text{EFC}_{pt^2}^-$, which is $(2n+1)/7$ times larger than the original private key. New decryption algorithm

Algorithm 4.4: New decryption algorithm for $\text{EFC}_{pt^2}^-$

Input : $\mathbf{b} = (\theta_1, \dots, \theta_n)$, the private key $A, B, S \in \mathbb{F}^{n \times n}$ and $T \in \mathbb{F}^{2n \times 2n}$.

Output: The plaintext $\mathbf{z} \in \mathbb{F}^n$ s.t. $P(\mathbf{z}) = \mathbf{c}$.

- 1 $\Theta \leftarrow \mathbf{b}^\top \cdot \mathbf{b}$, $(\Theta_1, \dots, \Theta_n) \leftarrow \varphi^{-1}(\Theta)$
- 2 Define $\Lambda \in \mathbb{F}^{n \times n}$, where $\Lambda_i = \varphi^{-1}(\theta_i^4)$
- 3 $M \leftarrow S(A - B)\Lambda$
- 4 **for** $i \leftarrow 1$ **to** n **do**
- 5 $N^{(i)} \leftarrow T^{-1}(SB\Theta_i || SA\Theta_i)^\top \in \mathbb{F}^{2n \times n}$
- 6 **for** $j \leftarrow 1$ **to** $2n$ **and** $i \leftarrow 1$ **to** n **do**
- 7 $U_i^{(j)} \leftarrow N_j^{(i)}$
- 8 $L \leftarrow \sum_{i=1}^{2n-a} c_i U^{(i)} - M^\top$
- 9 **for** $\mathbf{v} = (v_1, \dots, v_a) \in \mathbb{F}^a$ **do**
- 10 $H \leftarrow L + \sum_{i=1}^a v_i U^{(2n-a+i)}$;
- 11 $\mathbf{ker} \leftarrow \text{RightKer}(H)$
- 12 **for** $\mathbf{z} \in \mathbf{ker}$ **do**
- 13 **if** $P(\mathbf{z}) = \mathbf{c}$ **then**
- 14 **Return** \mathbf{z}
- 15 **break**

for $\text{EFC}_{pt^2}^-$ works similarly to that of EFC_p^- , detailed procedures are shown in Algorithm 4.4.

We can analyze the complexity of the decryption algorithm for EFC_{pt^2} using the same approach as in the proof of Proposition 4.4.1. The number of field operations involved in the new decryption algorithm for $\text{EFC}_{pt^2}^-$ is

$$2^{(a-1)} \left(\frac{14}{3}n^3 + \left(\frac{11}{2} - 2a \right)n^2 - \left(a + \frac{19}{6} \right)n + a \right) + 4n^3 - 2an^2. \quad (4.20)$$

4.4.3 Implementation

We verify the effectiveness of our decryption method through implementation operated on a 2.10 GHz Intel[®] Xero[®] Gold 6130 CPU with Magma V2.23-10 under originally claimed 80-bit security parameters, see [SDP16], and then compare the results with complexity given in (4.5), (4.10), (4.14) and (4.20). The implementation results are given in Table 4.1.

Table 4.1: Timing comparison between old EFC_p^- , $\text{EFC}_{pt^2}^-$ with new EFC_p^- , $\text{EFC}_{pt^2}^-$ under 80-bit security parameter given in [SDP16], $[+, \times]$ represents the number of involved field operations

	Scheme (n, a)	KeyGen.[s]	Enc.[s]	Dec.[s]	$[+, \times]_{\mathbb{F}}$
Old	$\text{EFC}_p^-(83, 10)$	0.022	0.00023	0.116	2.96×10^9
	$\text{EFC}_{pt^2}^-(83, 8)$	0.023	0.00023	0.028	1.18×10^9
New	$\text{EFC}_p^-(83, 10)$	0.022	0.00023	0.013	1.32×10^9
	$\text{EFC}_{pt^2}^-(83, 8)$	0.023	0.00023	0.003	0.33×10^9

From Table 4.1, we know, under 80-bit security parameter given in [SDP16], theoretically our new decryption algorithms are 2.24 times faster for EFC_p^- and 3.58 times faster for $\text{EFC}_{pt^2}^-$ than the original decryption algorithms, and the speed-up obtained in our implementation are 8.92 and 9.33 times for EFC_p^- and $\text{EFC}_{pt^2}^-$, respectively.

Since our proposed decryption algorithms do not alter public keys for EFC_p^- and $\text{EFC}_{pt^2}^-$, their security does not change. As for the private key, to match with our proposed decryption algorithms, we use new private keys, which is $2n/7$ times larger for EFC_p^- , and $(2n+1)/7$ times larger for $\text{EFC}_{pt^2}^-$ compared to the original private keys.

4.5 Hybrid attack against EFC

Because of the minus modifier, the most efficient attack against EFC_p^- and $\text{EFC}_{pt^2}^-$ is expected to be the algebraic attack, see Section 4.2.3, which computes the Gröbner basis of the ideal generated by the public key and the field equations. Normally we

use F4/F5 [Fau99, Fau02] algorithms, which has complexity given in equation (4.3). In [SDP16], upper bounds for the degree of regularity of EFC_p^- and $\text{EFC}_{pt^2}^-$ are given:

$$d_{reg} \leq \frac{r}{2} + 2, \quad r = \begin{cases} 2 + a, & \text{EFC}_p^-, \\ 4 + a, & \text{EFC}_{pt^2}^-. \end{cases} \quad (4.21)$$

And [SDP16] also claimed the degree of regularity of EFC_p^- and $\text{EFC}_{pt^2}^-$ lie close to these upper bounds. Under the assumption that these upper bounds are identical to the degree of regularity of EFC_p^- and $\text{EFC}_{pt^2}^-$, 80-bit and 128-bit security parameters are estimated in [SDP16] and [WIDT18], respectively.

In this section, we apply hybrid algebraic attack on EFC_p^- and $\text{EFC}_{pt^2}^-$ to learn how parameters n and a affect their degree of regularity. All of our experiments are conducted on a 2.10 GHz Intel[®] Xero[®] Gold 6130 Processor with Magma V2.23-10, where F4 algorithm is implemented.

4.5.1 Notations

Notations used in this section are as follows:

- d_{reg} : degree of regularity (see Section 4.2.3).
- step_{deg} : step degree, a sequence of degrees of polynomials appeared in all the steps of F4 algorithm.
- s_{deg} : solving degree, the degree of the most time-consuming step of F4 algorithm.
- e : the number of variables evaluated in hybrid algebraic attack (see Section 4.2.3).
- time/gb : average time cost of one round of F4 algorithm in hybrid algebraic attack.
- total time (est.) : estimated time cost of hybrid algebraic attack, and it holds the following equation: $\text{total time (est.)} = 2^{e-1} * (\text{time/gb})$.
- total time : average time cost of hybrid algebraic attack.
- s, h, d, y : second, hour, day, year.

4.5.2 Hybrid attack on EFC_p^- and update secure parameters

For EFC_p^- , we first verify if the claimed 80-bit security parameter in [SDP16] indeed has 80-bit security against hybrid attack. Let e be a positive integer. We guess values for variables (x_1, \dots, x_e) from $\mathbb{F}^e (|\mathbb{F}^e| = 2^e)$, and evaluate the system G , see (4.2), with those

guessing values, then perform algebraic attack on the obtained system. The experiment terminates when a solution can be found. The results are shown in Table 4.2. From this table, we know when $e = 16$, both of d_{reg} and s_{deg} are 4, and the complexity of the hybrid attack is around $2^{16} \binom{83-16+4-1}{4}^2 \binom{83-16}{2} \approx 2^{67}$ by formula (4.3). Therefore, the originally proposed 80-bit security parameter fails in achieving its claimed security level.

Table 4.2: Hybrid algebraic attack on $\text{EFC}_p^-(83, 10)$

e	step_{deg}	d_{reg}	s_{deg}	time/gb	total time (est.)
15	(2,3,4,4,5,...)	4	≥ 5	—	—
16	(2,3,4,4,4,4)	4	4	2.618d	235.065y
17	(2,3,4,4,4)	4	4	1.896d	340.418y
18	(2,3,4,4,3)	4	4	23.658h	353.983y
19	(2,3,4,4)	4	4	4.900h	146.647y
20	(2,3,4,4)	4	4	4.016h	240.370y
21	(2,3,4,4)	4	4	3.132h	374.864y

To update secure parameters for EFC_p^- , we need to find a lower bound for the degree of regularity for EFC_p^- . The upper bound given in (4.21) is given following analysis on HFE based schemes [DH11, DY13, DK12, DG10], and it is deduced according to the fact that the d_{reg} of a subsystem is equal to or larger than the full polynomial system. Following this approach, we know the d_{reg} of an EFC_p^- system with $a - 1$ polynomial deleted is equal to or smaller than that of an EFC_p^- system with a polynomial deleted. It means a lower bound for d_{reg} of EFC_p^- can be obtained as long as one linear combination of the polynomials deleted by minus modifier can be recovered. However, this indicates the total break of EFC_p^- , and it is a very difficult task. Therefore, we experiment with small parameters to find an experimental lower bound. Parameter n is fixed to be 41, then we experiment with parameter a increasing. The results are shown in Table 4.3, from which we know there are two turning points for the degree of regularity. One of them is when a rises to 11 from 10, the degree of regularity turns into 5, the other one is when a rises from 17 to 18, the degree of regularity turns into 6. We regard them as lower bounds to update new parameters in Section 4.5.4. Since the d_{reg} of an EFC_p^- system with larger parameter is equal to or larger than the d_{reg} of an EFC_p^- system with smaller parameter, updating new parameters using those experimental lower bounds

should be enough.

Table 4.3: Behave of d_{reg} of $\text{EFC}_p^-(n = 41, a)$ with $e = 1$

a	step_{deg}	d_{reg}	s_{deg}	total time
10	(2,3,4,4,5,2,3,4,5,6,7)	4	5	2.181h
11	(2,3,4,5,2,3,4,5,6,7)	5	5	2.550h
12	(2,3,4,5,4,2,3,4,5,6,7)	5	5	2.707h
13	(2,3,4,5,5,2,3,4,6,7)	5	5	6.204h
14	(2,3,4,5,5,2,3,4,5,6,7)	5	5	6.995h
15	(2,3,4,5,5,2,3,4,5,6,7)	5	5	7.911h
16	(2,3,4,5,5,2,3,4,5,6,7)	5	5	8.713h
17	(2,3,4,5,5,2,3,4,5,6,7)	5	5	9.305h
18	(2,3,4,5,6,...)	≥ 6	≥ 6	–

4.5.3 Hybrid attack on $\text{EFC}_{pt^2}^-$

We apply hybrid attack on $\text{EFC}_{pt^2}^-$ in the same approach as EFC_p^- , the results are shown in Table 4.4. This table shows hybrid attack cannot work any better than algebraic attack. Therefore, the originally proposed 80-bit security parameter set for $\text{EFC}_{pt^2}^-$ remains secure against hybrid attack. Table 4.4 also shows the variation of d_{reg} of $\text{EFC}_{pt^2}^-$ is more drastic compared to EFC_p^- , which is also the reason why we were not able to perform hybrid attack with small parameters. Even with $n = 40, a = 8$, which is expected to have $d_{reg} \leq 8$, it takes significantly long time to perform F4 algorithm that we were not able to get the results. We therefore use bound given in (4.21) to estimate 128-bit security level parameter.

From Table 4.4, we can see that there is a tendency of hybrid attack not outperforming direct attack for $\text{EFC}_{pt^2}^-$. Because of this, the estimation of secure parameters for $\text{EFC}_{pt^2}^-$ for high security levels strongly relies on a good theoretical estimation of degree of regularity. Therefore, the tightness of bound given in (4.21) should be further investigated, and we would like to continue working in this regard in the future.

Table 4.4: Hybrid algebraic attack on $\text{EFC}_{pt^2}^-(83, 8)$

e	step_{deg}	d_{reg}	s_{deg}	complexity
36	(2,3,4,5,6,...)	≥ 6	≥ 6	$\geq 2^{95}$
37	(2,3,4,5,5)	5	5	2^{89}
44	(2,3,4,5)	5	4	2^{93}
45	(2,3,4,4)	4	4	2^{88}
53	(2,3,4)	4	3	2^{92}
54	(2,3,3)	3	3	2^{87}

4.5.4 Update new secure parameters

Taking the experimental results of hybrid algebraic attack on EFC_p^- and $\text{EFC}_{pt^2}^-$ into account, we update their secure parameters, see Table 4.5. Old 128-bit security parameters are deduced according to (4.21), and new parameter for EFC_p^- are obtained by considering the experimental lower bound of degree of regularity in Section 4.5.2 and Section 4.5.3.

Table 4.5: Updated parameters sets

	Security	Old parameters	New parameters
EFC_p^-	80-bit	(83, 10)	(241, 11)
	128-bit	(467, 10)	(1523, 18)
$\text{EFC}_{pt^2}^-$	80-bit	(83, 8)	(83, 8)
	128-bit	(467, 8)	(467, 8)

4.5.5 Implementation under new parameters

New secure parameters for EFC_p^- are updated, we verify its performance by implementation, and compare it with $\text{EFC}_{pt^2}^-$, see Table 4.6. From this table, we can easily see, $\text{EFC}_{pt^2}^-$ performs better when it comes to efficiency.

Table 4.6: Performance comparison between EFC_p^- and $\text{EFC}_{pt^2}^-$ with updated parameters and new decryption algorithms

	Scheme (n, a)	KeyGen.[s]	Enc.[s]	Dec.[s]
80-bit	$\text{EFC}_p^-(241, 11)$	0.352	0.0013	0.090
	$\text{EFC}_{pt^2}^-(83, 8)$	0.023	0.00023	0.003
128-bit	$\text{EFC}_p^-(1523, 18)$	403.946	0.215	562.435
	$\text{EFC}_{pt^2}^-(467, 8)$	3.734	0.0072	0.065

4.6 Conclusion

We have shown that EFC_p^- and $\text{EFC}_{pt^2}^-$ both contain redundant computation in their decryption. By removing them, both of their decryption process can be improved. Based on this idea, we proposed our decryption algorithms for EFC_p^- and $\text{EFC}_{pt^2}^-$ without weakening their security. We also showed originally proposed 80-bit security parameter for EFC_p^- failed in achieving the claimed security level through hybrid algebraic attack. However, the originally proposed 80-bit security parameter for $\text{EFC}_{pt^2}^-$ seemed secure enough. We estimated new secure 80-bit and 128-bit security parameter for EFC_p^- and compared its performance with $\text{EFC}_{pt^2}^-$ under the same security level, and conclude that it is recommended to use $\text{EFC}_{pt^2}^-$ since 128-bit EFC_p^- is very inefficient.

Moreover, a thorough investigation on degree of regularity of EFC_p^- and $\text{EFC}_{pt^2}^-$ is inadequate. With a good estimation of degree of regularity, parameter choosing process will become simpler comparing to our method of using experimental results on hybrid attack. We will continue working on this in the future.

Chapter 5

On the algebraic aspects of solving the minrank problem

The minrank problem is often considered in the cryptanalysis of multivariate cryptography 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 chapter, we review two existing methods, the Kipnis-Shamir method (KS), and minors modeling for solving a minrank instance, and then propose a mixed method that merges these two methods. Our method uses a bilinear subsystem from the KS method and a subsystem from minors modeling. It is at least as effective as the KS method, and does not require as many minors as minors modeling. Moreover, we consider applying the hybrid approach on multivariate polynomials solved in our mixed method to further improve our method. We then revisit the minrank attack on Rainbow and conclude the previous complexity analysis of the minrank attack on Rainbow is overestimated, and provide the correct complexity of the minrank attack on NIST PQC 2nd round Rainbow parameters.

The result in this chapter was published on the proceedings of the conference *The 21st World Conference on Information Security Application* under the title “Revisiting the Minrank Attack on Multivariate Cryptography” [WINT20].

5.1 Introduction

With currently widely used cryptosystems, RSA [RSA78] and ECC [Kob87], being threatened by the development of quantum computers because of Shor's quantum algorithm [Sho97], research on the post-quantum cryptography has become more urgent. NIST [AASA⁺18, CJL⁺16] 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 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 rank metric decoding problem, which is the main problem considered in code-based cryptography, can be reduced to the minrank problem as well. The minrank problem ($\text{MR}(q, n, m, r)$) asks one to find a linear combination of given $m + 1$ matrices M_0, M_1, \dots, M_m over a finite field of order q that has rank between 1 and r . This problem is proven to be an NP-complete problem [BFS99], and in the field of multivariate cryptography, by far there are three different methods proposed for solving it, that are the Kipnis-Shamir (KS) method [KS99], minors modeling [BFP13] and linear algebra search method [GC00b].

In multivariate cryptography, many attempts on building secure cryptosystems failed due to their weakness against the minrank attack, for example, HFE [KS99], SRP [PPST18], ZHFE [CSTV17], and TTM [GC00b]. Techniques such as enlarging parameters or applying modifiers are applied to some multivariate cryptosystems such as Rainbow [DS05] and HFEv- [PCG01b, PCY⁺15] 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 KS method until Verbel et al. [VBC⁺19] gave their analysis. They gave a method of constructing non-trivial syzygies (see definition in §5.2.1) for super-determined minrank instances, and hence understanding the first fall degree (see definition in §5.2.1) of the polynomial system obtained from the KS method, which indicates a tighter complexity bound. This result is used on cryptanalysis on rank metric code-based cryptosystems [BBB⁺20, BBC⁺20]. As its advantage, the KS method gives low first fall degrees for super-determined minrank instances. As its drawback, it introduces many new variables and its analysis on using

subsystems are not thorough. On the other hand, various analyses on the complexity of minors modeling are given [CG19, FDS10, FDS13]. This method does not introduce new variables but requires a heavy computation of many minors.

Contribution

The first contribution of this chapter is to propose a new method of solving the minrank problem called the mixed method. It uses a bilinear subsystem (say S_μ) from the KS method and a subsystem (say T_μ) from minors modeling. When S_μ is an under-determined subsystem, adding T_μ to S_μ means adding more equations to S_μ without introducing any new variables, and it can possibly decrease the overall degree of regularity. Conversely, adding S_μ to T_μ significantly reduces the number of spurious solutions of T_μ . Therefore, when S_b is under-determined, the proposed mixed method possibly improves the KS method. When S_μ is an over-determined subsystem, T_μ is not needed, which means our mixed method reduces to the KS method.

Another contribution of this chapter is to consider applying the hybrid approach [BFP09] on multivariate polynomials solved in the mixed method. The hybrid approach is a combination of exhaustive search and Gröbner basis computation for solving a set of multivariate polynomials. The values of a few variables of a polynomial system are specified randomly before solving this system, the process terminates once the correct values are used. A bilinear subsystem is used in the mixed method, which means there are two sets of variables. The more significant set of variables being specified expects to bring more degree drops in the first fall degree of a polynomial system. For the mixed method, specifying every variable from the set that has fewer variables, according to our experiments, expects to decrease its first fall degree by 1. We also revisit the minrank attack on NIST PQC 2nd Rainbow proposal by considering the KS, minors modeling and the mixed method all together. We find that the previous complexity analysis of the minrank attack on Rainbow Ia, IIIc and Vc parameters [DCP⁺17b] are overestimated, that are $2^{156.1}$, $2^{578.0}$ and $2^{771.7}$. We update the new complexity to be $2^{138.1}$, $2^{308.1}$, $2^{405.4}$, and our investigation shows that the proposed parameters for Rainbow are secure from the minrank attack.

This chapter is organized as follows. Section 5.2 explains about multivariate quadratic problem and bilinear systems. In section 5.3, we review the minrank problem, the KS method and minors modeling. In section 5.4, we propose a mixed method for solving the

minrank problem and discuss the behavior of the mixed method coupling with the hybrid approach. In section 5.5, we present experimental results on scaled-down Rainbow and application of our method on Rainbow. Finally, section 5.6 gives a conclusion.

5.2 The MQ problem and bilinear systems

5.2.1 Multivariate quadratic problem

Let \mathbb{F} be a finite field of order q , $m, n \in \mathbb{N}$, and $R := \mathbb{F}[x_1, \dots, x_n]$ be the polynomial ring in variables x_1, \dots, x_n over \mathbb{F} . Multivariate quadratic problem (MQ problem) is defined in Definition 2.1.1. An effective method for solving this problem is through Gröbner basis computation [Buc65]. Efficient algorithms for computing a Gröbner basis include XL [CKPS00], F4 [Fau99] and F5 [Fau02]. A good indicator of the complexity of computing a Gröbner basis is the *degree of regularity* (d_{reg}) [BFSY05], which is the maximal polynomial degree appeared during a process of computing a Gröbner basis (see Definition 3.3.15). This complexity mainly comes from a computation of the row echelon form of a Macaulay matrix of degree d_{reg} . Suppose such a Macaulay matrix has size $R_{d_{reg}} \times C_{d_{reg}}$, then the complexity of the fast algorithm proposed in [Sto10] for computing its row echelon form is given by $O(R_{d_{reg}} C_{d_{reg}}^{\omega-1})$, where $2 \leq \omega \leq 3$ is the linear algebra constant. The degree of regularity d_{reg} for random systems can be precisely evaluated, but hard to estimate for specific families of polynomial systems. Therefore, in cryptographical studies, d_{reg} is often approximated by the *first fall degree* (d_{ff}). To define the first fall degree, we need to be familiar with a notion called non-trivial syzygies.

Definition 5.2.1 (Syzygy). Let $\{h_1, \dots, h_m\} \in R$ be a set of polynomials. A syzygy of (h_1, \dots, h_m) is an m -tuple $(s_1, \dots, s_m) \in R^m$ such that $\sum_{i=1}^m s_i h_i = 0$. The degree of a syzygy $\mathbf{s} = (s_1, \dots, s_m)$ is defined as $\deg(\mathbf{s}) = \max_{1 \leq i \leq m} \deg(s_i h_i)$.

The linear combinations of m -tuples $(s_1, \dots, s_m) \in R^m$ with $s_i = h_j, s_j = -h_i$ for some i, j ($i \neq j$) and $s_t = 0$ for $t \neq i, j$ are called *trivial syzygies*. The syzygies that are not linear combinations of the trivial syzygies are called *non-trivial syzygies*. Non-trivial syzygies of the homogeneous components of the highest degree of h_1, \dots, h_m account for the non-trivial degree falls during a Gröbner basis computation.

Definition 5.2.2 (First fall degree d_{ff}). Let $\{f_1, \dots, f_m\} \subset R$ be a set of polynomials and $\{\tilde{f}_1, \dots, \tilde{f}_m\} \subset R$ be their homogeneous component of the highest

degree. Its first fall degree is the smallest degree d_{ff} such that there exist non-trivial syzygies $(s_1, \dots, s_m) \in R^m$ of $(\tilde{f}_1, \dots, \tilde{f}_m)$ with $\max_i(\deg(s_i \tilde{f}_i)) = d_{ff}$, satisfying $\deg(\sum_{i=1}^m s_i f_i) < d_{ff}$ but $\sum_{i=1}^m s_i \tilde{f}_i \neq 0$.

Many results on multivariate cryptosystems are based on analyzing d_{ff} [DK12, DH11, DY13, DG10], although it is not always true that d_{ff} and d_{reg} are very close, experimental and theoretical evidences in these results have shown it seems to be true for some cryptographic schemes.

5.2.2 Bilinear system

A bilinear polynomial is defined as follows.

Definition 5.2.3 (Bilinear polynomial). Let $\mathbf{x} = (x_1, \dots, x_{n_1})$, $\mathbf{y} = (y_1, \dots, y_{n_2})$ 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 linear 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 Jacobian matrices to explain these properties. The Jacobian matrix of a set of bilinear polynomials is defined as follows.

Definition 5.2.4 (Jacobian matrix). Given a set of bilinear polynomials $F = (f_1, \dots, f_m) \in \mathbb{F}[\mathbf{x}, \mathbf{y}]^m$, then the Jacobian matrices of F with respect to variables \mathbf{x} and \mathbf{y} are given by

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

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

Proposition 5.2.1. Let $F = (f_1, \dots, f_m) \in \mathbb{F}[\mathbf{x}, \mathbf{y}]^m$ be a set of bilinear polynomials. For $G = (g_1, \dots, g_m) \in \mathbb{F}[\mathbf{y}]^m$, it is a syzygy of F if $G \cdot \text{Jac}_{\mathbf{x}}(F) = 0$. Moreover, if G is non-zero, then G is a non-trivial syzygy of F . Similar statement holds for $\text{Jac}_{\mathbf{y}}(F)$.

Proof. By the definition of the Jacobian matrix, we have $\text{Jac}_{\mathbf{x}}(F)\mathbf{x} = F^\top$. Given $G \cdot \text{Jac}_{\mathbf{x}}(F) = 0$, we easily obtain $\sum_{i=1}^m g_i f_i = G \cdot \text{Jac}_{\mathbf{x}}(F)\mathbf{x} = 0$. Therefore, G is a syzygy, and it also lies in the left kernel of $\text{Jac}_{\mathbf{x}}(F)$.

Since the trivial syzygies of F contains variables \mathbf{x} and \mathbf{y} , and G can only contain

variables \mathbf{x} , we know if G is non-zero, it is a non-trivial syzygy.

Similar proof can be applied to $\text{Jac}_{\mathbf{y}}(F)$ case. \square

From the above proposition, we can construct some non-trivial syzygies of a set of homogeneous bilinear polynomials $F = (f_1, \dots, f_m) \in \mathbb{F}[\mathbf{x}, \mathbf{y}]^m$ using its Jacobian matrices, which have linear polynomials as its entries, and we need to compute their left kernels. By Cramer's rule, see [FSEDS11], we know the kernel of such matrices have elements in the span of its maximal minors (also see example 5.2.1 and 5.2.2 in appendix). Here, maximal minor refers to determinants of square submatrices with the maximal size of a matrix.

Example 5.2.1. Let \mathbb{Q} be the field of rational numbers. We consider solving

$$\begin{bmatrix} a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0 \text{ for } x_1, x_2, x_3 \text{ over the field } \mathbb{Q}(a_1, a_2, a_3, b_1, b_2, b_3).$$

We convert it to the echelon form : $\begin{bmatrix} a_1 & a_2 & a_3 \\ 0 & \frac{b_2 a_1 - b_1 a_2}{a_1} & \frac{b_3 a_1 - b_1 a_3}{a_1} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = 0$. Let $x_3 = t$ for any $t \in \mathbb{Q}$ then $x_2 = -t \left(\frac{b_3 a_1 - b_1 a_3}{b_2 a_1 - b_1 a_2} \right)$, $x_1 = t \left(\frac{b_2 a_3 - b_3 a_2}{b_2 a_1 - b_1 a_2} \right)$. 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 5.2.2. Consider solving $\begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ b_1 & b_2 & b_3 & b_4 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0$ for x_1, \dots, x_4 over the field $\mathbb{Q}(a_1, a_2, a_3, a_4, b_1, b_2, b_3, b_4)$.

We convert it to the echelon form : $\begin{bmatrix} a_1 & a_2 & a_3 & a_4 \\ 0 & \frac{b_2 a_1 - b_1 a_2}{a_1} & \frac{b_3 a_1 - b_1 a_3}{a_1} & \frac{b_4 a_1 - b_1 a_4}{a_1} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = 0$.

Let $x_3 = t, x_4 = s$ for any $t, s \in \mathbb{Q}$. Then we have $x_1 = -\begin{vmatrix} a_2 & a_4 \\ b_2 & b_4 \end{vmatrix} s - \begin{vmatrix} a_2 & a_3 \\ b_2 & b_3 \end{vmatrix} t$, $x_2 =$

$$\begin{vmatrix} a_1 & a_4 \\ b_1 & b_4 \end{vmatrix} s + \begin{vmatrix} a_1 & a_3 \\ b_1 & b_3 \end{vmatrix} t, \quad x_3 = - \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} t, \quad x_4 = - \begin{vmatrix} a_1 & a_2 \\ b_1 & b_2 \end{vmatrix} s.$$

5.3 The minrank problem

In this section, we introduce the minrank problem and two existing methods for solving the minrank problem, the KS method and minors modeling.

5.3.1 The minrank problem

The minrank problem is defined as follows.

Problem 5.3.1 (Minrank Problem). Given a field \mathbb{F} of order q , a positive integer $r \in \mathbb{N}$ and $n \times n$ matrices $M_0, M_1, \dots, M_m \in \mathbb{F}^{n \times n}$, find $x_1, \dots, x_m \in \mathbb{F}$ such that $\Delta = M_0 + \sum_{i=1}^m x_i M_i$, $0 < \text{Rank}(\Delta) \leq r$. A minrank instance is denoted by $\text{MR}(q, n, m, r)$.

5.3.2 Minors Modeling

Minors modeling [BFP13] is based on the fact that all $(r+1) \times (r+1)$ minors of $\Delta = M_0 + \sum_{i=1}^m x_i M_i$ vanish at (x_1, \dots, x_m) when Δ has rank no larger than r . This method gives 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 [CG19, FDS10, FDS13], intensive analyses on the property of the ideal generated by polynomials from minors modeling are given. Minors modeling, as its advantage, does not introduce any other variables except x_1, \dots, x_m . But it requires a computation of as many as $\binom{n}{r+1}^2$ minors of a matrix with linear polynomial entries. If we consider the XL algorithm, there are $\binom{m+r+1}{r+1}$ monomials up to degree $r+1$, and we need $\binom{m+r+1}{r+1}$ independent equations to terminate the XL algorithm, which means the complete $\binom{n}{r+1}^2$ equations are unnecessary to achieve d_{reg} being $r+1$ when $\binom{n}{r+1}^2 > \binom{m+r+1}{r+1}$. Every minor is a degree $r+1$ polynomial in variables x_1, \dots, x_m , which has $\binom{m+r+1}{r+1}$ terms at most. Suppose obtaining every coefficient takes complexity $O(1)$, computing $\min\{\binom{n}{r+1}^2, \binom{m+r+1}{r+1}\}$ minors requires a complexity of $O\left(\min\{\binom{n}{r+1}^2, \binom{m+r+1}{r+1}\} \cdot \binom{m+r+1}{r+1}\right)$. Note that when partial minors are used, d_{reg} turns to be higher and spurious solutions appear. Moreover, solving the polynomials obtained from making those minors vanish takes complexity $O\left(\binom{m+r+1}{r+1}^\omega\right)$, where $2 \leq \omega \leq 3$ is a linear algebra constant.

5.3.3 The Kipnis-Shamir method

The KS method [KS99] was first used to break the HFE cryptosystem [Pat96]. This method is based on the fact that the dimension of the right kernel of $M_0 + \sum_{i=1}^m x_i M_i$ should be no smaller than $n - r$, since it has rank no larger than r . There exists a canonical echelonized basis for this right kernel, we put these basis vectors into a matrix as column vectors, then this matrix should be in the form of $\begin{bmatrix} I_{n-r} \\ K \end{bmatrix}$, where I_{n-r} is the identity matrix of size $n - r$ and K is an $r \times (n - r)$ matrix. We denote the column vectors of $\begin{bmatrix} I_{n-r} \\ K \end{bmatrix}$ by $\hat{\mathbf{k}}_1, \hat{\mathbf{k}}_2, \dots, \hat{\mathbf{k}}_{n-r}$. Then we have

$$\Delta \begin{bmatrix} I_{n-r} \\ K \end{bmatrix} = \Delta [\hat{\mathbf{k}}_1, \hat{\mathbf{k}}_2, \dots, \hat{\mathbf{k}}_{n-r}] = 0. \quad (5.1)$$

If we regard the entries of K as new variables:

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

then we obtain a system of $n(n - r)$ bilinear equations in variables $\mathbf{x} = (x_1, \dots, x_n)$ and $\mathbf{k} = (k_1, k_2, \dots, k_{r(n-r)})$ from (5.1). Moreover, $n - r$ subsystems can also be obtained from (5.1), which are denoted by S_1, S_2, \dots, S_{n-r} as follows:

$$\underbrace{\Delta \cdot \hat{\mathbf{k}}_1}_{S_1} = \mathbf{0}, \quad \underbrace{\Delta \cdot [\hat{\mathbf{k}}_1 \quad \hat{\mathbf{k}}_2]}_{S_2} = \mathbf{0}, \quad \dots, \quad \underbrace{\Delta \cdot [\hat{\mathbf{k}}_1 \quad \hat{\mathbf{k}}_2 \quad \cdots \quad \hat{\mathbf{k}}_{n-r}]}_{S_{n-r}} = \mathbf{0}.$$

Solving a subsystem may take less time than solving the full system. However, the d_{ff} of a subsystem is no smaller than the d_{ff} of the full system, i.e. $d_{ff}(S_i) \geq d_{ff}(S_{n-r})$ for $i = 1, \dots, n - r - 1$. In [VBC⁺19], this is pointed out and they suggest using subsystems that are determined or over-determined since under-determined subsystems tend to have higher d_{ff} and give spurious solutions.

In [VBC⁺19], Verbel et al. gave a tight bound on the d_{ff} of S_{n-r} , and they experimentally showed the d_{reg} is close to their bound as well. As its advantages, the KS method can construct a polynomial system more easily compared to minors modeling, optionally determined or over-determined subsystems can be used, and for super-determined

minrank instances, over-determined subsystems from the KS method have low d_{ff} . However, this method introduces more variables than minors modeling, i.e. variables $k_1, k_2, \dots, k_{r(n-r)}$. Moreover, precise bounds on the d_{ff} of the subsystems S_1, \dots, S_{n-r-1} are not yet clear. According to [VBC⁺19], when S_{n-r} is used, by only multiplying monomials in variables from \mathbf{k} in the XL algorithm, a complexity of $O\left(\binom{\mu r + d_{ff}}{d_{ff}}^\omega\right)$ can be achieved, where $d_{ff} = \min_{1 \leq d \leq r-1} \left\{ d \mid \binom{r}{d}n > \binom{r}{d+1}m \right\} + 2$. And with high probability, d_{ff} remains the same when S_μ is used, where $\max\{\frac{m}{n-r}, d_{ff} - 1\} \leq \mu \leq n - r$.

5.4 Our proposed method

In this section, we propose a new method that combines the KS method and minors modeling.

5.4.1 The mixed method

Let $\mathbf{b}_1, \dots, \mathbf{b}_n$ be the row vectors of Δ , i.e.

$$\Delta = [\mathbf{b}_1^\top \quad \mathbf{b}_2^\top \quad \cdots \quad \mathbf{b}_{n-r}^\top \quad \mathbf{b}_{n-r+1}^\top \quad \mathbf{b}_{n-r+2}^\top \quad \cdots \quad \mathbf{b}_n^\top]^\top.$$

Since the rank of a matrix is the maximal number of linearly independent column vectors, we assume the last r rows $\mathbf{b}_{n-r+1}, \dots, \mathbf{b}_n$ are linearly independent. Then $\{\mathbf{b}_i, \mathbf{b}_{n-r+1}, \dots, \mathbf{b}_n\}$ for each $i = 1, \dots, n-r$ is linearly dependent, which gives us in total $n-r$ linear relations. We can translate the linear dependence of $\{\mathbf{b}_i, \mathbf{b}_{n-r+1}, \dots, \mathbf{b}_n\}$ into either “find k_j for $\sum_{j=1}^r k_j \mathbf{b}_{n-r+j} = \mathbf{b}_i$ ” or “ $(r+1) \times (r+1)$ minors of the matrix $[\mathbf{b}_i^\top \quad \mathbf{b}_{n-r+1}^\top \quad \cdots \quad \mathbf{b}_n^\top]^\top$ vanish.” The approach where new variables k_i are introduced corresponds to a subsystem in the KS method.

Let $1 \leq \mu \leq n-r$ be an integer, in the mixed method, we first realize the linear dependence of $\{\mathbf{b}_i, \mathbf{b}_{n-r+1}, \dots, \mathbf{b}_n\}$ for $i = 1, \dots, \mu$ by introducing new variables, the resulting polynomial system is the same with S_μ in the KS method. Then we compute $(r+1) \times (r+1)$ minors of the matrices $[\mathbf{b}_i^\top \quad \mathbf{b}_{n-r+1}^\top \quad \cdots \quad \mathbf{b}_n^\top]^\top$ for $i = \mu+1, \dots, n-r$, we denote this system as T_μ . Finally we solve the S_μ and T_μ combined polynomial system.

As shown in [VBC⁺19], the more kernel vectors $\hat{\mathbf{k}}_1, \hat{\mathbf{k}}_2, \dots, \hat{\mathbf{k}}_{n-r}$ are used in the KS method, the smaller its d_{ff} and d_{reg} will become, and when a subsystem S_μ is over-determined, its d_{ff} is smaller than $r+2$. In the mixed method, the S_μ and T_μ combined polynomial system is used, and polynomials in T_μ have degree $r+1$, which means adding

T_μ to an over-determined S_μ does not reduce the overall d_{ff} . Hence we only use an under-determined S_μ in our mixed method, i.e. $\mu = 1, \dots, \lfloor \frac{m}{n-r} \rfloor$. The motivation of our method is, on one hand, adding T_μ to an under-determined S_μ to make all subsystems in the KS method usable and substantially reduces spurious solutions of those under-determined subsystems. On the other hand, mixing two methods to achieve the lowest d_{ff} possible without introducing many additional variables and computing many minors.

5.4.2 Complexity analysis

In this subsection, we investigate the complexity of the mixed method.

I. Case $\mu = 1$

Since $F_1 = S_1 \cup T_1$, where S_1 is a bilinear polynomial system and T_1 has polynomials of degree $r + 1$. We first analyze the first fall degree of S_1 , and we have the following proposition.

Proposition 5.4.1. Let S_1^h be the homogeneous components of the highest degree of S_1 . S_1^h has non-trivial syzygies in variables \mathbf{k} of degree $m + 2$ and non-trivial syzygies in variables \mathbf{x} of degree $r + 2$.

Proof. Let S_1^h be the degree two homogeneous components of S_1 , then the lowest degree of its non-trivial syzygies coincide with the d_{ff} of S_1 . The left kernel of $\text{Jac}_{\mathbf{k}}(S_1^h)$ gives non-trivial syzygies of S_1^h in variables \mathbf{x} . Since $\text{Jac}_{\mathbf{k}}(S_1^h)$ is an $n \times r$ matrix, and has maximal minors of degree r , we know it gives us non-trivial syzygies of degree $r + 2$. On the other hand, the left kernel of $\text{Jac}_{\mathbf{x}}(S_1^h)$ gives non-trivial syzygies of S_1^h in variables \mathbf{k} . Since $\text{Jac}_{\mathbf{x}}(S_1^h)$ is an $n \times m$ matrix, and it has maximal minors of degree $\min\{m, n\}$, and gives non-trivial syzygies of degree $\min\{m + 2, n + 2\}$. □

From Proposition 2, we have the first fall degree d_{ff} of S_1 is no larger than $\min\{r + 2, m + 2\}$. Furthermore, we know the left kernel of $\text{Jac}_{\mathbf{k}}(S_1^h)$ (resp. $\text{Jac}_{\mathbf{x}}(S_1^h)$) are n -tuples with polynomial entries, which can be computed from the maximal minors of $\text{Jac}_{\mathbf{k}}(S_1^h)$ (resp. $\text{Jac}_{\mathbf{x}}(S_1^h)$), if there exist common divisors among those polynomial entries, we would have non-trivial syzygies with lower degrees. These common divisors are difficult to compute mathematically, but can be confirmed using experiments. We found that when $n \geq m + r$ holds, such common divisors appear, which means when $n \geq m + r$,

there exists non-trivial syzygies of degree $\leq \min\{r + 1, m + 1\}$. Related experimental results are shown in Table 5.1, it verifies the correctness of using $\text{Jac}_{\mathbf{x}}(S_1^h)$ and $\text{Jac}_{\mathbf{k}}(S_1^h)$ to analyze the d_{ff} of S_1 , and also confirms the existence of the aforementioned common divisors when $n \geq m + r$.

Table 5.1: Experiments on the d_{ff} and d_{reg} of S_1 , and degrees of the non-trivial syzygies of S_1^h from the left kernel of $\text{Jac}_{\mathbf{x}}(S_1^h)$ and $\text{Jac}_{\mathbf{k}}(S_1^h)$. $\ker(\text{Jac}_{\mathbf{x}}(S_1^h))$ (resp. $\ker(\text{Jac}_{\mathbf{k}}(S_1^h))$) are computed on Magma using the function “Kernel”, where the $F4$ algorithm is used. Note that when $n \geq m + r$ satisfies, we have $d_{ff} = d_{reg}$

(q, n, m, r)	$\ker_{\text{left}}(\text{Jac}_{\mathbf{x}}(S_1^h))$	$\ker_{\text{left}}(\text{Jac}_{\mathbf{k}}(S_1^h))$	$d_{ff}(S_1)$	$d_{reg}(S_1)$
$(7, 6, 5, 3)$	7	5	5	7
$(7, 7, 5, 3)$	7	5	5	7
$(7, 8, 5, 3)$	5	4	4	4
$(7, 9, 5, 3)$	4	3 or 4	3 or 4	4
$(7, 10, 5, 3)$	3 or 4	3	3	3
$(7, 11, 5, 3)$	3	3	3	3

Therefore, we conclude with the following upper bounds for the d_{ff} of S_1 :

$$\begin{aligned}
n < m + r, & \quad d_{ff}(S_1) \leq \min\{r + 2, m + 2\}, \\
n = m + r, & \quad d_{ff}(S_1) \leq \min\{r + 1, m + 1\}, \\
n > m + r, & \quad d_{ff}(S_1) < \min\{r + 1, m + 1\}.
\end{aligned} \tag{5.2}$$

Regarding the first fall degree of F_1 and its relation to the first fall degree of S_1 , we have the following proposition.

Proposition 5.4.2. Let $d_{ff}(S_1)$ be the first fall degree of S_1 , which depends on either the non-trivial syzygies in variables \mathbf{k} of degree no larger than $m + 2$ or non-trivial syzygies in variables \mathbf{x} of degree no larger than $r + 2$. From these non-trivial syzygies of S_1 , non-trivial syzygies of F_1 can be constructed, and the first fall degree of F_1 is no larger than $d_{ff}(S_1)$.

Proof. We will discuss this in two cases according to whether any non-trivial syzygies exist in $\mathcal{I}_R = \langle F_1 \rangle$. If F_1 is regular or semi-regular (see [BFSY05]), then it does not have non-trivial syzygies, its d_{reg} and d_{ff} are the same. Subsystem S_1 of F_1 can be regarded

as deleting a few polynomials from F_1 , which will either increase its d_{reg} or not change it.

Suppose (s_1, \dots, s_n) is a non-trivial syzygy of degree d . Then we can construct a syzygy of degree d for F_1^h , which is $(s_1, \dots, s_n, 0, \dots, 0)$, where F_1^h consists of the homogeneous components of the highest degree of F_1 .

If (s_1, \dots, s_n) is a non-trivial syzygy in variables \mathbf{k} , $s_i (1 \leq i \leq n)$ are polynomials of degree no larger than m , then $(s_1, \dots, s_n, 0, \dots, 0)$ is a non-trivial syzygy of F_1^h by Proposition 5.2.1. On the other hand, if (s_1, \dots, s_n) is in variables \mathbf{x} , $s_i (1 \leq i \leq n)$ will have degree no larger than r . Since T_1 consists of polynomials of degree $r+1$, we know $(s_1, \dots, s_n, 0, \dots, 0)$ can only be a non-trivial syzygy of F_1^h by Proposition 5.2.1. According to the definition of d_{ff} , we know the d_{ff} of F_1 is at most d since there may exist other non-trivial syzygies of F_1^h that have a smaller degree than d . Therefore, the statement is proved. \square

According to this proposition, we have Equation (5.2) holds also for F_1 .

II. Case $\mu = 2, \dots, \lfloor \frac{m}{n-r} \rfloor$

When $1 < \mu \leq n - r$, our method solves a polynomial system $F_\mu = S_\mu \cup T_\mu$, where S_μ is an under-determined bilinear system and T_μ consists of polynomials of degree $r+1$. Similarly, an upper bound for the d_{ff} of F_μ can be obtained by analyzing the first fall degree d_{ff} of S_μ and T_μ . Let S_μ^h be the degree two homogeneous components of S_μ , then we have

$$\begin{aligned} \text{Jac}_{\mathbf{k}}(S_\mu^h) &= I_\mu \otimes \text{Jac}_{\mathbf{k}}(S_1^h), \\ \text{Jac}_{\mathbf{x}}(S_\mu^h) &= \left(I_n \otimes \begin{bmatrix} k_1 & k_2 & \cdots & k_r \\ k_{r+1} & k_{r+2} & \cdots & k_{2r} \\ \vdots & \vdots & \ddots & \vdots \\ k_{(\mu-1)r+1} & k_{(\mu-1)r+2} & \cdots & k_{\mu r} \end{bmatrix} \right) \cdot L^\mu, \end{aligned} \quad (5.3)$$

where $L^\mu \in \mathbb{F}^{nr \times m}$ is a matrix derived from the matrices M_1, \dots, M_m .

By (5.3), the non-trivial syzygies from the left kernel of $\text{Jac}_{\mathbf{k}}(S_\mu^h)$ and $\text{Jac}_{\mathbf{k}}(S_1^h)$ should have same degree, so $\text{Jac}_{\mathbf{k}}(S_\mu^h)$ gives non-trivial syzygies of degree $r+2$ when $n < m+r$ and less than or equal to r when $n \geq m+r$. Similarly, the left kernel of $\text{Jac}_{\mathbf{x}}(S_\mu)$ also gives non-trivial syzygies of S_μ . But analyzing their precise degree is difficult as aforementioned common divisors have to be analyzed. Nevertheless, we know the d_{ff} and d_{reg} of S_μ should be decreasing with μ increasing from 1 to $\lfloor \frac{m}{n-r} \rfloor$ since S_μ becomes

less under-determined. Therefore, the d_{ff} of the mixed method of $\mu = 2, \dots, \lfloor \frac{m}{n-r} \rfloor$ is upper bounded by the first fall degree of the mixed method of $\mu = 1$ given in (5.2).

5.4.3 Further Improvement

In this section, we consider applying the hybrid approach [BFP09] on the mixed method. That is to exhaustively guess a few variables before applying Gröbner basis computation algorithms on the polynomial system obtained by the mixed method. The question here is to guess which variables. In both the KS method and the mixed method, we have bilinear systems, which means there are two sets of different variables. We want to find the set of variables to guess that minimizes the total complexity. Table 5.2 presents results about applying the hybrid approach on S_1 under $(q, n, m, r) = (7, 13, 8, 5)$, which should have $d_{ff} \leq 6$ because of the non-trivial syzygies from $\text{Jac}_{\mathbf{k}}(S_1^h)$. Note that specifying variables from \mathbf{x} does not change the degree of polynomials in T_1 , therefore its first fall degree will always be no smaller than $r + 1$.

Table 5.2: Results of hybrid approach on S_1 under $(q, n, m, r) = (7, 13, 8, 5)$, and $\ker(\text{Jac}_{\mathbf{x}}(S_1^h))$ (resp. $\ker(\text{Jac}_{\mathbf{k}}(S_1^h))$) are computed on Magma using the function “Kernel”, where the $F4$ algorithm is used. $\lfloor x \rfloor$ means the nearest integer to x .

# variables specified in \mathbf{x}	0	1	2	3	4	5	6	7	8	$0 \leq i \leq m$
deg of syzygies from $\ker_{\text{left}}(\text{Jac}_{\mathbf{x}}(S_1^h))$	8	6	5	4	3	3	2	2	-	$\approx 8 - \lfloor \frac{m \cdot i}{m} \rfloor$
deg of syzygies from $\ker_{\text{left}}(\text{Jac}_{\mathbf{k}}(S_1^h))$	6	5	5	4	3	3	2	2	-	$\approx 6 - \lfloor \frac{r \cdot i}{m} \rfloor$
d_{ff}	6	5	5	4	3	3	2	2	1	$\approx 6 - \lfloor \frac{r \cdot i}{m} \rfloor$

# variables specified in \mathbf{k}	0	1	2	3	4	5	$0 \leq j \leq r$
deg of syzygies from $\ker_{\text{left}}(\text{Jac}_{\mathbf{x}}(S_1^h))$	8	6	5	3	2	-	$\approx 8 - \lfloor \frac{m \cdot j}{r} \rfloor$
deg of syzygies from $\ker_{\text{left}}(\text{Jac}_{\mathbf{k}}(S_1^h))$	6	5	4	3	2	-	$\approx 6 - \lfloor \frac{r \cdot j}{r} \rfloor$
d_{ff}	6	5	4	3	2	1	$\approx 6 - \lfloor \frac{r \cdot j}{r} \rfloor$

The table tells us specifying every variable from \mathbf{k} (resp. \mathbf{x}) brings -1 to the degree of syzygies from the left kernel of $\text{Jac}_{\mathbf{k}}(S_1^h)$ (resp. $\text{Jac}_{\mathbf{x}}(S_1^h)$), and this can be rationalized as this specification changes the size of $\text{Jac}_{\mathbf{k}}(S_1^h)$ (resp. $\text{Jac}_{\mathbf{k}}(S_1^h)$) to $n \times (r - 1)$ (resp. $n \times (m - 1)$), which gives non-trivial syzygies of 1 less degree. Moreover, specifying every variable from \mathbf{k} (resp. \mathbf{x}) decreases approximately the degrees of the non-trivial syzygies

from $\text{Jac}_{\mathbf{x}}(S_1^h)$ (resp. $\text{Jac}_{\mathbf{k}}(S_1^h)$) by $\lfloor \frac{m}{r} \rfloor$ (resp. $\lfloor \frac{r}{m} \rfloor$). Note that this technique can be applied to the mixed method and the KS method.

Practically, $F_\mu = S_\mu \cup T_\mu$ is used in the mixed method, T_μ is for decreasing the overall d_{ff} and reducing spurious solutions. When either variables in \mathbf{x} or \mathbf{k} are specified, S_μ may turn into a less under-determined or an over-determined system. In this case, less polynomials or no polynomials from T_μ are needed. Moreover, specifying either m variables from \mathbf{x} or r variables from \mathbf{k} leads to a complete solve of a minrank instance.

Summarizing the discussion, we assume using XL algorithm and only multiplying by monomials from variables \mathbf{k} or \mathbf{x} , let $d_{\mathbf{x}}$ (resp. $d_{\mathbf{k}}$) be the lowest degree of the non-trivial syzygies from $\text{Jac}_{\mathbf{x}}(S_\mu^h)$ (resp. $\text{Jac}_{\mathbf{k}}(S_\mu^h)$), then the complexity of our mixed method is bounded by

$$\begin{aligned}
 m \geq r \quad & O \left(\min_{1 \leq k < r} \left\{ q^k \cdot \left(\min \left\{ \binom{r\mu - k + d_{\mathbf{x}} - \lfloor \frac{km}{r} \rfloor}{d_{\mathbf{x}} - \lfloor \frac{km}{r} \rfloor} \right\}^\omega, \binom{m + d_{\mathbf{k}} - k}{d_{\mathbf{k}} - k}^\omega \right\} + knm \right) \right\} \\
 m < r \quad & O \left(\min_{1 \leq k < r} \left\{ q^k \cdot \left(\min \left\{ \binom{m - k + d_{\mathbf{k}} - \lfloor \frac{rk}{m} \rfloor}{d_{\mathbf{k}} - \lfloor \frac{rk}{m} \rfloor} \right\}^\omega, \binom{r\mu + d_{\mathbf{x}} - k}{d_{\mathbf{x}} - k}^\omega \right\} + kn^2 \right) \right\} \right)^{*1}
 \end{aligned}$$

where $2 \leq \omega \leq 3$ is the linear algebra constant. Note that the complexity given above are for polynomial solving only. For computing minors, suppose S_μ after specifying k variables is under-determined, $(n - r - \mu) \binom{n}{r+1}$ minors will be used in the mixed method, which requires a complexity of $O \left((n - r - \mu) \binom{n}{r+1} \binom{m - k + r + 1}{r+1} \right)$ when k variables from \mathbf{x} are specified, and $O \left((n - r - \mu) \binom{n}{r+1} \binom{m + r + 1}{r+1} \right)$ when k variables from \mathbf{k} are specified. This computation of minors can be done in parallel and its complexity is neglectable compared to that of polynomial solving.

5.5 Experiments and application

5.5.1 Experiments

The parameters we choose to run experiments on proportionally coincide with Rainbow [DS05], which is $(q, v, o_1, o_2) = (16, 5, 5, 5)$. Note that the first layer of Rainbow

*1 Note that the computation involving kn^2 and kmn can be done in parallel.

central map polynomials have rank $v + o_1$, and second layer polynomials have full rank. Its public key has $o_1 + o_2$ polynomials. Therefore, we can recover some first layer Rainbow central map polynomials by solving some minrank instances $\text{MR}(q, v + o_1 + o_2, o_1 + o_2, v + o_1)$. However, the span of the low rank polynomials hidden in Rainbow, say \mathcal{S}_c (dimension o_1), is a subspace of the span of the public key, say \mathcal{S}_p (dimension $o_1 + o_2$). The intersection of \mathcal{S}_c with any dimension $o_2 + 1$ subspace of \mathcal{S}_p is a subspace of dimension no smaller than 1. Therefore, using $o_2 + 1$ polynomials p_1, \dots, p_{o_2+1} from the public key of Rainbow, we are able to recover partial Rainbow secret key by solving $\text{MR}(q, v + o_1 + o_2, o_2 + 1, v + o_1)$. Moreover, if we fix the variable x_1 from x_1, \dots, x_m in the minrank problem to be 1, with probability $\frac{q-1}{q}$, we can still obtain a solution. Therefore, breaking Rainbow is almost equivalent to solving $\text{MR}(q, v + o_1 + o_2, o_2, v + o_1)$. All of our experiments are executed on a 2.10 GHz Intel[®] Xeon[®] Gold 6130 Processor with Magma V2.24-8 [BCP97], where F4 algorithm [Fau99] is implemented. We run 5 experiments for each set of parameter.

Table 5.3: Experimental results on $\text{MR}(16, 15, 5, 10)$, which is equivalent to breaking $\text{Rainbow}(q, v, o_1, o_2) = (16, 5, 5, 5)$. The best complexity is $2^{23.4}$, which is when we use S_1 with hybrid approach of specifying 3 variables out of x_1, \dots, x_m . $d_{\mathbf{x}}$ (resp. $d_{\mathbf{k}}$) denotes the lowest degree of the non-trivial syzygies derived from the Jacobian matrix of S_1 w.r.t variables \mathbf{x} (resp. \mathbf{k}), and t denotes the total time for computing minors and solving the obtained polynomials with $F4$ algorithm

MR(16, 15, 5, 10)							
method		d_{ff}	d_{reg}	$d_{\mathbf{k}}$	$d_{\mathbf{x}}$	t [s]	Complexity ($\omega = 2.8$)
minors		11	11	—	—	*2	$\binom{m+r+1}{r+1}^\omega = \binom{5+11}{11}^\omega \approx 2^{33.9}$
KS	S_2	5	5	—	—	615.57	$\binom{2 \cdot r + 5}{5}^\omega = \binom{2 \cdot 10 + 5}{5}^\omega \approx 2^{44.0}$
	S_3	4	4	—	—	30.49	$\binom{3 \cdot r + 4}{4}^\omega = \binom{3 \cdot 10 + 4}{4}^\omega \approx 2^{43.4}$
New S_1		6	6	$10 = r$	6	67.20	$\min\{\binom{m+10}{10}^\omega, \binom{r+6}{6}^\omega\} \approx 2^{32.3}$
	fix x_1	5	5	8	5	10.80	$q \cdot \left(\min\{\binom{m-1+10-2}{10-2}^\omega, \binom{r+6-1}{6-1}^\omega\} + n^2 \right) \approx 2^{29.1}$
	fix x_1, x_2	4	4	6	4	5.12	$q^2 \cdot \left(\min\{\binom{m-2+10-4}{10-4}^\omega, \binom{r+6-2}{6-2}^\omega\} + 2n^2 \right) \approx 2^{25.9}$
	fix x_1, \dots, x_3	3	3	4	3	4.73	$q^3 \cdot \left(\min\{\binom{m-3+10-6}{10-6}^\omega, \binom{r+6-3}{6-3}^\omega\} + 3n^2 \right) \approx 2^{23.4}$
	fix x_1, \dots, x_4	2	2	2	2	4.34	$q^4 \cdot \left(\min\{\binom{m-4+10-8}{10-8}^\omega, \binom{r+6-5}{6-5}^\omega\} + 4n^2 \right) \approx 2^{25.8}$
	fix x_1, \dots, x_5	—	—	—	—	—	$q^m \cdot (5n^2 + \frac{n^3}{6}) \approx 2^{30.7}$

Table 5.3 shows results on breaking $\text{Rainbow}(16, 5, 5, 5)$ by solving a minrank instance $\text{MR}(q, n, m, r) = \text{MR}(16, 15, 5, 10)$. Since $m + r = n$ satisfies, S_1 in KS and the mixed

method is determined, no extra minors are needed. Hence, in the mixed method, we only consider using S_1 coupling with the hybrid approach. Namely, we randomly specify variables from x_1, \dots, x_m in S_1 , and try to solve S_1 . Note for this scaled down Rainbow parameter with $m \leq r$ and $n = m + r$, when only S_1 is used, the degree of the non-trivial syzygies from the Jacobian matrix of S_1 w.r.t variables \mathbf{x} is $r = 10$, and the degree of the non-trivial syzygies from the Jacobian matrix of S_1 w.r.t variables \mathbf{k} is $m + 1 = 6$. Comparing to the KS method and minors modeling, specifying 3 variables in S_1 gives the best complexity, $2^{23.4}$.

To testify that our mixed method is indeed efficient, we also conduct experiments on $\text{MR}(16, 9, 6, 6)$ and $\text{MR}(16, 11, 6, 8)$. The results are shown in Table 5.4. For minors modeling and the mixed method, computing minors are necessary, and they are computed on Magma in our experiments, the timings are recorded under the label t_{minors} in Table 5.4. t_{F4} means timings for polynomial solving using $F4$ algorithm with graded reverse lexicographical monomial order. This table shows that minors modeling requires a long time on computing all the minors needed, but takes shorter time on polynomial solving compared to the KS and mixed method. As for the KS method, computations of minors are not required, but it can take a considerably long time on polynomial solving for some minrank instances, such as parameters presented in Table 5.4. As for the mixed method, not as many minors as minors modeling are needed, and it is faster in polynomial solving than the KS method for certain parameters such as ones shown in Table 5.4.

5.5.2 Application on Multivariate Cryptography

Rainbow

A public key from $\text{Rainbow}(q, v, o_1, o_2)$ gives an $\text{MR}(q, n, m, r) = \text{MR}(q, v + o_1 + o_2, o_2, v + o_1)$. For example, $\text{Rainbow}(16, 32, 32, 32)$, which achieves NIST type I security, gives us $\text{MR}(q, n, m, r) = \text{MR}(16, 96, 32, 64)$. If we use minors modeling, d_{ff} is estimated to be 65, assuming $\omega = 2.8$ gives us a complexity $\binom{m+r+1}{r+1}^\omega = 2^{238.5}$. Note that computing minors has a complexity of $\min\left\{\binom{m+r+1}{r+1}^2, \binom{n}{r+1}^2 \binom{m+r+1}{r+1}\right\} = \min\{2^{170.39}, 2^{337.59}\} = 2^{170.39}$. If we use KS method considering [VBC⁺19] with $n - r = 32$ kernel vectors, d_{ff} is estimated to be 18 and we assume using $d_{ff} - 1 = 17$ out of 32 kernel vectors and mul-

^{*2} Due to the limited computation resources, we were not able to obtain this timing. The computation of minors did not finish in 2 days, which is the maximal time limit for our platform.

Table 5.4: Experimental results on solving minrank instances with minors modeling (see §5.3.2), the KS method (see §5.3.3) and the mixed method (see §5.4.1)

(q, n, m, r) (16, 9, 6, 6)	method		d_{ff}	d_{reg}	t_{minors} [s]	t_{F4} [s]	$t_{\text{minors}} + t_{F4}$ [s]
	minors		7	7	39.01	1.56	40.57
	KS	S_2	5	6	0	234.42	234.42
		S_3	4	5	0	114.29	114.29
	mixed	$\mu = 1$	8	8	2.17	8.54	10.71
		$\mu = 2$	5	6	1.08	247.88	248.96

(q, n, m, r) (16, 11, 6, 8)	method		d_{ff}	d_{reg}	t_{minors} [s]	t_{F4} [s]	$t_{\text{minors}} + t_{F4}$ [s]
	minors		9	9	1183.59	14.40	1197.99
	KS	S_2	6	6	0	13375.86	13375.86
		S_3	5	5	0	3296.61	3296.61
	mixed	$\mu = 1$	8	10	43.04	395.57	438.61
		$\mu = 2$	6	6	21.52	16823.13	16844.65

tiply only by monomials from kernel variables in the XL algorithm still has $d_{ff} = 18$, then we have a complexity $\binom{17r+d_{ff}}{d_{ff}}^\omega = 2^{362.0}$. Using exhaustive search on variables x_1, \dots, x_m and verifying the solution cost a complexity of $q^m \cdot \left(mn^2 + \frac{n^3}{6}\right) \approx 2^{146.8}$, here mn^2 accounts for the computation of $\sum_{i=1}^m x_i M_i$, which can be done in parallel and $\frac{n^3}{6}$ accounts for verifying the rank of $M_0 + \sum_{i=1}^m x_i M_i$ using Gaussian elimination. Since for the given minrank instance, $m + r = n$ satisfies, we only need to use S_1 in the mixed method. Similar to the results in Table 5.3, by specifying $k = 30$ variables from x_1, \dots, x_m , non-trivial syzygies from $\text{Jac}_{\mathbf{k}}(S_1)$ will have degree $64 - 30 \cdot 2 = 4$, namely we have $d_{\mathbf{k}} = 4$, which gives us a complexity of $q^k \cdot \left(\binom{m-k+r-2k}{r-2k}^\omega + kn^2\right) \approx 2^{138.1}$. It is much lower than the claimed value $2^{156.1}$ presented in NIST PQC Rainbow proposal [DCP⁺17b] (see Table 5.5).

For parameter IIIc, Rainbow(256, 68, 36, 36), there is a minrank instance $\text{MR}(q, n, m, r) = \text{MR}(256, 140, 36, 104)$. Exhaustive search on variables x_1, \dots, x_m and verifying the correctness of the solution require a complexity $q^m \cdot \left(mn^2 + \frac{n^3}{6}\right) = 2^{308.1}$. Minors modeling has a complexity $\binom{m+r+1}{r+1}^\omega \approx 2^{313.2}$. When $n - r = 36$ kernel vectors are used in the KS method, its d_{ff} is expected to be 23. Assuming using $d_{ff} - 1 = 22$ kernel vectors also has $d_{ff} = 23$ gives a complexity of $\binom{22r+23}{23}^\omega \approx 2^{510.7}$. When the mixed method is used, since $m + r = n$ satisfies, we only need to use S_1 . The non-trivial syzygies from $\text{Jac}_{\mathbf{x}}(S_1)$

have degree $m + 1$ and ones from $\text{Jac}_{\mathbf{K}}(S_1)$ have degree r considering experiment results in Table 5.3. Moreover, since the cardinality of the field is 256, applying the hybrid approach will not bring any benefit. Therefore, the mixed method has a complexity of $\min\left\{\binom{m+r}{r}^\omega, \binom{r+m+1}{m+1}^\omega\right\} \approx 2^{312.0}$, which is also much lower than the claimed complexity $2^{578.0}$ given in [DCP⁺17b].

Similarly, for parameter Vc, Rainbow(256, 92, 48, 48), its complexities of minrank attack using exhaustive search on variables x_1, \dots, x_m and verifying the correctness of the solution, minors modeling, the KS method and the mixed method are $2^{405.4}$, $2^{421.7}$, $2^{705.8}$ and $2^{420.5}$, respectively.

Table 5.5: Complexity of the minrank attack on NIST PQC standardization 2nd round Rainbow proposal with different methods, minrank exhaustive represents the attack that exhaustively searches the values of x_1, \dots, x_m , and verify whether the solution gives a matrix of the target rank

security	(q, v, o_1, o_2)	$\text{MR}(q, n, m, r)$	complexity in [DCP ⁺ 17b]	minrank exhaustive	minors	KS	mixed
Ia	(16, 32, 32, 32)	(16, 96, 32, 64)	$2^{156.1}$	$2^{146.8}$	$2^{238.5}$	$2^{362.0}$	$2^{138.1}$
IIIc	(256, 68, 36, 36)	(16, 140, 36, 104)	$2^{578.0}$	$2^{308.1}$	$2^{313.2}$	$2^{510.7}$	$2^{312.0}$
Vc	(256, 92, 48, 48)	(256, 188, 48, 140)	$2^{771.7}$	$2^{405.4}$	$2^{421.7}$	$2^{705.8}$	$2^{420.5}$

From the above-mentioned discussions, we know the previous complexity analysis on the minrank attack presented in the NIST PQC standardization 2nd round Rainbow proposal is overestimated, but the minrank attack is not enough to break Rainbow.

5.6 Conclusion

In this chapter, methods for solving the minrank problem were considered. We reviewed two of the existing methods, the KS method and minors modeling, and some results on their complexities. We proposed a mixed method that combined the KS method and minors modeling. The new system used an under-determined bilinear subsystem from the KS and a subsystem from the minors modeling. When the bilinear subsystem is under-determined, the mixed method possibly outperforms the KS method and minors modeling. When the bilinear subsystem is over-determined, the mixed method has the same complexity as the KS method.

We also considered applying the hybrid approach on multivariate polynomials solved in our mixed method. A bilinear subsystem is used in the mixed method, so we considered specifying the set of variables that could minimize the complexity, which is the set that had fewer variables, and every variable specified at least reduced the first fall degree by 1. Finally we revisit the minrank attack on NIST PQC 2nd round Rainbow proposal, and found that originally estimated complexities of the minrank attack on Rainbow Ia, IIIc and Vc, which are $2^{156.1}$, $2^{578.0}$, $2^{771.7}$, are overestimated. We updated them to be $2^{138.1}$, $2^{308.1}$, $2^{405.4}$, and concluded Rainbow is secure from the minrank attack.

Chapter 6

Algebraic cryptanalysis of multivariate encryption scheme PERN

The security of multivariate public-key cryptography is based on the hardness of solving the MQ problem. In [Yas18], a new variant of the MQ problem called the constrained MQ problem is considered using on constructing multivariate encryption schemes. This method is called *pq-method*. In this chapter, we consider a new trapdoor which is similar to the pq-method and can be used on constructing secure multivariate encryption schemes, and we evaluate its security by considering algebraic attacks.

This chapter is based on the result published in the section 5.5 of the paper at conference *Post-Quantum Cryptography – PQCrypto 2020* titled “Multivariate Encryption Schemes Based on Polynomial Equations over Real Numbers” [YWT20]. My contribution to this work is the security analysis and secure parameter estimation of the new scheme called PERN via algebraic techniques, which can be found in section 6.3 and 6.4.

Throughout this section, \mathbb{F}_q, \mathbb{Z} denotes a finite field of q elements and integers respectively, $\mathbf{x} = (x_1, \dots, x_n)$ are n variables over \mathbb{F}_q or \mathbb{Z} , $\mathbb{F}_q[\mathbf{x}]$ and $\mathbb{Z}[\mathbf{z}]$ are polynomial rings.

6.1 Introduction

The MQ problem, defined in Definition 2.1.1, is a NP-complete problem that serves as a pillar that supports the security of multivariate public key cryptography. Its variant, the constraint MQ problem, defined as

Definition 6.1.1 (Constraint MQ problem). Given a set of polynomials $F = (f_1, \dots, f_m) \in \mathbb{F}_q[\mathbf{x}]^m$ and an integer L , find $\mathbf{z} \in \mathbb{Z}$ such that

$$f_1(\mathbf{z}) = \dots = f_m(\mathbf{z}),$$

$$-\frac{L}{2} \leq z_i \leq \frac{L}{2} \quad (i = 1, \dots, n),$$

was used to construct multivariate encryption schemes in [Yas18] for the first time. The constraint MQ problem is very similar to short solutions to nonlinear equations problem (SSNE problem) proposed in [SP17], hence its hardness is also related to the short integer solution problem (SIS problem) in lattice [Ajt96].

Since most of the proposed multivariate encryption scheme over the years such as MI [MI88], HFE [Pat96], ZHFE [PBD14], EFC [SDP16], Square [CBD⁺09] and SRP [YS16] are broken, developing secure multivariate encryption schemes have always been a topic in multivariate public-key cryptography. Currently, there are few schemes that remain secure are simple matrix scheme [TDTD13], HFERP [IPST⁺18] and EFLASH [CST18], which are all proposed quite recently. If we take a close look at the multivariate encryption schemes proposed in the past, all of them are broken because of some special structures hidden in their private keys. To avoid this, we consider using completely random private keys and sacrificing its efficiency performances slightly in exchange for high security. This is the basic idea behind polynomial equations over the real numbers (PERN), which is a new multivariate encryption scheme.

PERN, just like conventional multivariate encryption scheme, constructs its public keys from invertible affine maps and invertible quadratic polynomial maps. However, unlike other multivariate encryption schemes, it uses random quadratic polynomials. In general, multivariate encryption schemes use an injective polynomial map as part of its private key, but constructing an invertible injective map is not easy. PERN manipulates its plaintext and ciphertext space to make this randomly chosen polynomial map invertible and enable decryption. Identical to other multivariate schemes, algebraic attack can be applied to PERN. However, its plaintext and ciphertext are manipulated. When considering the complexity of the algebraic attack, this manipulation should be taken into consideration as well. In this chapter, we will investigate the security of PERN via algebraic techniques.

This chapter is organized as follows. In section 6.2, we recall the construction of pq method and PERN. Section 6.3 gives the algebraic cryptanalysis of PERN. In section 6.4,

we give secure parameters of PERN taking algebraic attack into consideration. Section 6.5 concludes this chapter.

6.2 Multivariate encryption scheme PERN

In this section, we recalled the construction designs for multivariate encryption schemes proposed in [Yas18] and [YWT20].

6.2.1 pq method

A standard multivariate encryption scheme generate key pairs as follows:

Private key It consists of three maps.

- Two invertible affine maps $S : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n, T : \mathbb{F}_q^m \rightarrow \mathbb{F}_q^m$.
- An easy-to-invert injective quadratic polynomial map $F : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^m$.

Public key A composition of three private maps, namely,

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

Encryption Given a plaintext $\mathbf{m} \in \mathbb{F}_q^n$, its corresponding ciphertext can be obtained by computing $\mathbf{c} = P(\mathbf{m})$.

Decryption Given a ciphertext $\mathbf{c} \in \mathbb{F}_q^m$, its corresponding plaintext can be obtained through recursively inverting private maps, namely

$$\mathbf{m} = S^{-1}(F^{-1}(T^{-1}(\mathbf{c}))).$$

The most important map of a multivariate encryption scheme is the map F , which is often called the *central map*. In [Yas18], a new type of trapdoor based on the constrained MQ problem is proposed, and we want to generalize its construction. The construction of the pq method is described as follows.

Notation Let p be a small prime, define $I_p := (-\frac{p}{2}, \frac{p}{2}] \cap \mathbb{Z}$ and let $L_p(a)$ be the least absolute remainder of a dividing p .

Private key It consists of three different maps.

- An easy-to-invert polynomial map that is constructed as follows:
 - Randomly choose a polynomial map $\tilde{F} = (\tilde{f}_1, \dots, \tilde{f}_n) \in \mathbb{Z}[\mathbf{x}]^n$, where f_i are polynomials with coefficients in I_p and $\tilde{F} \bmod p \in \mathbb{F}_p[\mathbf{x}]^n$ is injective and easy-to-invert. This polynomial map can be central maps of existing multivariate encryption schemes such as MI [MI88] and HFE [Pat96].
 - Randomly choose a polynomial map $\tilde{H} = (\tilde{h}_1, \dots, \tilde{h}_n) \in \mathbb{Z}[\mathbf{x}]^n$, where h_i are polynomials with coefficients in I_p .
 - Randomly choose $M, N \in \mathbb{N}$ such that

$$M \geq \max_{i=1, \dots, n} \{|\tilde{f}_i(\mathbf{a})| \mid \mathbf{a} \in I_p\},$$

$$N \geq \max_{i=1, \dots, n} \{|\tilde{h}_i(\mathbf{a})| \mid \mathbf{a} \in I_p\}.$$

- Randomly choose $r_1, \dots, r_n \in \mathbb{N}$ and a large prime q such that

$$r_i > 2M \ (i = 1, \dots, n) \text{ and } q > \max\{2r_i N + 2M \mid i = 1, \dots, n\}.$$
- Let $F := (\tilde{f}_1 + r_1 \tilde{h}_1, \dots, \tilde{f}_n + r_n \tilde{h}_n) \bmod q \in \mathbb{F}_q[\mathbf{x}]^n : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ be a central map.
- A permutation S of n elements.
- An invertible affine map $T : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$.

During the generation of map F , values r_1, \dots, r_n and q are chosen to guarantee inversion of F . Since $F = (\tilde{f}_1 + r_1 \tilde{h}_1, \dots, \tilde{f}_n + r_n \tilde{h}_n)$, and $(\tilde{f}_1 \bmod q, \dots, \tilde{f}_n \bmod q)$ is easy-to-invert, when inverting map F , we need to be able to separate the value of \tilde{F} and \tilde{H} . Figure 6.1 is a representation of $\tilde{F} + r\tilde{H}$. From this figure we have the following relations on M, N, r_i and q :

$$\begin{aligned} Nr_i + M &< \frac{q}{2}, \\ 2M &< r_i. \end{aligned} \tag{6.1}$$

$\{S, T, \tilde{F}, r_1, \dots, r_n\}$ is a private key for the pq method.

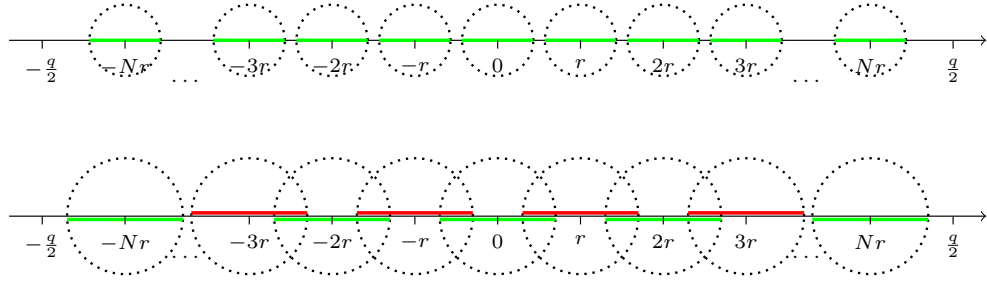


Figure 6.1: Representation of $\tilde{F} + r \cdot \tilde{H}$ with range $(-\frac{q}{2}, \frac{q}{2})$. The range of \tilde{F} and \tilde{H} are respectively $(-M, M)$ and $(-N, N)$. The center of circles represent the value of \tilde{H} , and radius of circles are M . Red and green segments represents possible values of $\tilde{F} + r\tilde{H}$. To guarantee the inversion of $\tilde{F} + r\tilde{H}$, segments can not overlap, which gives us the choices of r and q .

Public key A public key is given by a composition of T, F and S , which is a polynomial map:

$$P = T \circ F \circ S : I_p^n \rightarrow \mathbb{F}_q^n.$$

Encryption Given a message $\mathbf{m} \in I_p^n$, its corresponding ciphertext can be obtained by computing $\mathbf{c} = P(\mathbf{m})$.

Decryption Given a ciphertext $\mathbf{c} \in \mathbb{F}_q^n$ and a private key $\{S, T, \tilde{F}, r_1, \dots, r_n\}$, its corresponding plaintext can be obtained by performing the following steps:

- Invert the map T and compute $T^{-1}(\mathbf{c}) = \mathbf{z} \in \mathbb{F}_q^n$.
- Use r_1, \dots, r_n to find unique values \mathbf{z}' for \tilde{F} and solve the polynomial system

$$\tilde{f}_1 \bmod q = z'_1,$$

$$\vdots$$

$$\tilde{f}_n \bmod q = z'_n,$$

and let the solution be $\mathbf{w} \in \mathbb{F}_q^n$.

- Compute $(w'_1, \dots, w'_n) = (L_p(w_1), \dots, L_p(w_n)) \in I_p^n$.
- Invert the permutation S and compute $\mathbf{m} = S^{-1}(w'_1, \dots, w'_n) \in I_p^n$.

6.2.2 Construction of PERN

In the pq method, one easy-to-invert polynomial map and one random polynomial map are used in constructing its central map, which could possibly lead to weakness. Therefore, we drop the easy-to-invert polynomial and use two random polynomial maps to construct a trapdoor one-way function, which is proposed in [YWT20].

Notation Let p, \hat{p} be small primes, define $I_p := (-\frac{p}{2}, \frac{p}{2}] \cap \mathbb{Z}$, $I_{\hat{p}} := (-\frac{\hat{p}}{2}, \frac{\hat{p}}{2}] \cap \mathbb{Z}$ and let $L_p(a), L_{\hat{p}}(a)$ be the least absolute remainder of a dividing p and \hat{p} , respectively.

Private key The private key consists of two maps.

- An easy-to-invert polynomial map that is constructed as follows:
 - Randomly choose a polynomial map $\tilde{F} = (\tilde{f}_1, \dots, \tilde{f}_n) \in \mathbb{Z}[\mathbf{x}]^n$, where \tilde{f}_i are polynomials with coefficients in $I_{\hat{p}}$.
 - Randomly choose a polynomial map $\tilde{H} = (\tilde{h}_1, \dots, \tilde{h}_n) \in \mathbb{Z}[\mathbf{x}]^n$, where \tilde{h}_i are polynomials with coefficients in $I_{\hat{p}}$.
 - Randomly choose $M, N \in \mathbb{N}$ such that

$$M > \max_{i=1, \dots, n} \{|\tilde{f}_i(\mathbf{a})| \mid \mathbf{a} \in I_p\},$$

$$M > \max_{i=1, \dots, n} \{|\tilde{h}_i(\mathbf{a})| \mid \mathbf{a} \in I_p\}.$$

- Randomly choose $r_1, \dots, r_n \in \mathbb{N}$ and a large prime q such that

$$r_i > 2M \ (i = 1, \dots, n) \text{ and } q > \max\{2r_i N + 2M \mid i = 1, \dots, n\}.$$

- Let $F := (\tilde{f}_1 + r_1 \tilde{h}_1, \dots, \tilde{f}_n + r_n \tilde{h}_n) \bmod q \in \mathbb{F}_q[\mathbf{x}]^n : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$ be a central map.
- An invertible affine map $T : \mathbb{F}_q^n \rightarrow \mathbb{F}_q^n$.

$\{T, \tilde{F}, \tilde{H}, r_1, \dots, r_n\}$ serves as a private key.

Public key A composition of two maps given by

$$P = T \circ F : I_p^n \rightarrow \mathbb{F}_q^n.$$

Encryption Given a public key P and a plaintext $\mathbf{m} \in I_p^n$, its corresponding ciphertext can be obtained by computing

$$\mathbf{c} = P(\mathbf{m}).$$

Decryption Given a private key $\{T, \tilde{F}, \tilde{H}, r_1, \dots, r_n\}$ and a ciphertext $\mathbf{c} \in \mathbb{F}_q^n$, its corresponding plaintext can be obtained by performing the following steps:

- Invert the map T and compute $T^{-1}(\mathbf{c}) = \mathbf{z} \in \mathbb{F}_q^n$.
- Use r_1, \dots, r_n to uniquely find value $\hat{\mathbf{z}}$ and $\check{\mathbf{z}}$ for \tilde{F} and \tilde{H} , respectively.
- Solve the following system with a solution constraint in I_p^n :

$$\begin{cases} \tilde{f}_1 - \hat{z}_1 = 0, \\ \vdots \\ \tilde{f}_n - \hat{z}_n = 0, \\ \tilde{h}_1 - \check{z}_1 = 0, \\ \vdots \\ \tilde{h}_n - \check{z}_n = 0, \end{cases} \quad (6.2)$$

using algorithms from numerical analysis.

Remark 6.2.1. In the decryption process of PERN, numerical approach is used in solving nonlinear equations, which can be used for nonlinear equations of any degree. This implies the choice of central map F is not limited to quadratic polynomials.

6.2.3 Solving nonlinear systems with integer coefficients

In the decryption process of PERN, a polynomial system of $2n$ polynomials in n variables (Equation (6.2)) with a bound constraint on variables has to be solved. In this section, we consider methods for solving such polynomial systems.

Let $H = (h_1, \dots, h_{2n})$ be the polynomial system in Equation (6.2), it can be solved by transferring it to an optimization problem. Most modern and efficient algorithms for solving nonlinear systems of equations by minimizing a sum of squares. H can be

models as a least square problem as following:

$$\text{Minimize} \quad \theta(\mathbf{x}) = \frac{1}{2} \sum_{i=1}^{2n} h_i(\mathbf{x})^2. \quad (6.3)$$

To solve this optimization problem, we use a point sequence $\mathbf{x}_1, \mathbf{x}_2, \dots \in \mathbb{R}^n$ with a cluster point \mathbf{x}^* and \mathbf{x}_{k+1} is given by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + t_k \mathbf{d}_k,$$

where $\mathbf{d}_k \in \mathbb{R}^n$ is called a search direction that satisfies

$$\nabla \theta(\mathbf{x}_k) \mathbf{d}_k^\top = H(\mathbf{x}_k) J_H(\mathbf{x}_k) \mathbf{d}_k^\top < 0,$$

where $J_H(\mathbf{k})$ is the Jacobian matrix

$$J_H(\mathbf{x}_k) = \left(\begin{array}{ccc} \frac{\partial h_1}{\partial x_1} & \cdots & \frac{\partial h_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_{2n}}{\partial x_1} & \cdots & \frac{\partial h_{2n}}{\partial x_n} \end{array} \right) \bigg|_{\mathbf{x}=\mathbf{x}_k}$$

of H evaluated at point \mathbf{x}_k , and $t_k \in (0, 1)$ is called a step size, which satisfies the Armijo condition:

$$\forall \alpha \in (0, 1) \quad \theta(\mathbf{x}_k + t_k \mathbf{d}_k) - \theta(\mathbf{x}_k) \leq \alpha t_k H(\mathbf{x}_k) J_H(\mathbf{x}_k) \mathbf{d}_k^\top.$$

Eventually, the sequence $\mathbf{x}_1, \mathbf{x}_2, \dots$ globally converge to a cluster point \mathbf{x}^* that satisfies

$$\nabla \theta(\mathbf{x}^*) = H(\mathbf{x}^*) J_H(\mathbf{x}^*) = \mathbf{0}. \quad (6.4)$$

For H , the rank of $J_H(\mathbf{x}) \in \mathbb{R}^{2n \times n}$ is less than n , which means Equation (6.4) implies $H(\mathbf{x}^*) \in \ker(J_H(\mathbf{x}^*))$ and $H(\mathbf{x}^*)$ may not necessarily be $\mathbf{0}$. Therefore, we need to reselect sequences until $H(\mathbf{x}^*) = \mathbf{0}$ holds.

There are several methods proposed for choosing the search direction such as Newton's method and Levenberg-Marquardt method [Lev44, Mar63, Mor78]. However, Newton's method is known to have poor performances when solving overdetermined nonlinear equations, we hence use Levenberg-Marquardt method, which is available via the `fsolve(...)` command in MatlabTM.

Levenberg-Marquardt In the Levenberg-Marquardt method, the search direction \mathbf{d}_k is chosen to be

$$\mathbf{d}_k = -\nabla \theta(\mathbf{x}_k) (J_H(\mathbf{x})^\top J_H(\mathbf{x}))^{-1},$$

Algorithm 6.1: Decryption algorithm for PERN

Input : $H = (h_1, \dots, h_{2n}) \in \mathbb{Z}[x_1, \dots, x_n]^{2n}$, $\alpha, \beta, \gamma \in (0, 1)$ and p **Output:** A solution $\mathbf{z} \in \mathbb{Z}^n$ such that $z_i \in I_p$ for $i = 1, \dots, n$

```

1  $i \leftarrow 1$ 
2  $\mathbf{x}_i \leftarrow_R \left(-\frac{p}{2}, \frac{p}{2}\right]^n$ 
3 while  $i \neq 0$  do
4    $\mathbf{e}_i \leftarrow -H(\mathbf{x}_i)J_H(\mathbf{x}_i)$ 
5    $S_i \leftarrow J_H(\mathbf{x}_i)^\top J_H(\mathbf{x}_i)$ 
6    $\mathbf{d}_i \leftarrow S_i \mathbf{e}_i^{-1}$ 
7   Find  $l \in \mathbb{N}$  such that
      
$$\theta(\mathbf{x}_i + \beta^l \mathbf{d}_i) - \theta(\mathbf{x}_i) \leq -\alpha \beta^l \mathbf{e}_i \mathbf{d}_i^\top$$

8    $t_i \leftarrow \beta^l$ 
9    $\mathbf{x}_{i+1} \leftarrow \mathbf{x}_i + t_i \mathbf{d}_i$ 
10   $i \leftarrow i + 1$ 
11  if  $\|t_i \mathbf{d}_i\|_\infty < \gamma$  then
12    break
13  $\mathbf{x}'_i \leftarrow$  Round every entry in  $\mathbf{x}_i$  to its nearest integer.
14 if  $H(\mathbf{x}') = 0$  then
15   Return  $\mathbf{x}'$ 
16 else
17   Go back to step 1.

```

where $J_H(\mathbf{x})^\top J_H(\mathbf{x})$ is a positive definite symmetric matrix since $J_H(\mathbf{x}_k)$ is a $2n \times n$ matrix. The detailed description of this method is illustrated in Algorithm 6.1.

From the construction of PERN, we know a plaintext $\mathbf{m} \in I_p^n$ is a solution to H , but we need to discuss the possibilities of existing other solutions. There is in fact no theoretical facts to prove the number of solutions appeared using optimization techniques to solve H besides *Bézout's bound*, but this bound is not tight. However, experimentally we can always obtain only one solution, which is the plaintext.

6.3 Algebraic cryptanalysis of PERN

In this section, we investigate the security of PERN via algebraic techniques.

6.3.1 Exhaustive search

A plaintext of a PERN can be any elements in I_p , which has a cardinality p^n . Given a ciphertext $\mathbf{c} \in \mathbb{F}_q^n$ and a public key P , its corresponding plaintext \mathbf{m} that satisfies $P(\mathbf{m}) = \mathbf{c}$ can be exhaustively found in

$$O(p^n),$$

when a quantum computer is available, by running Grover's algorithm [Gro96] a complexity of $O(p^{\frac{n}{2}})$ can be achieved.

6.3.2 Algebraic attack

Given a ciphertext $\mathbf{c} \in \mathbb{F}_q^n$ and a public key $P = (p_1, \dots, p_n) \in \mathbb{F}_q[x_1, \dots, x_n]^n$, the algebraic attack solves the system

$$\begin{aligned} g_1 &= p_1 - c_1 = 0, \\ &\vdots \\ g_n &= p_n - c_n = 0, \end{aligned} \tag{6.5}$$

algebraically by hybrid approach [BFP09] of polynomial solving using XL algorithm [CKPS00] or Gröbner bases techniques such as F4 [Fau99] and F5 [Fau02] with exhaustive search. Moreover, since all variables are over I_p and p is a small prime, we can add the following trivial relations to Equation (6.5):

$$\begin{aligned} g_{n+1} &= \prod_{a \in I_p} (x_1 - a) = 0, \\ &\vdots \\ g_{2n} &= \prod_{a \in I_p} (x_n - a) = 0. \end{aligned} \tag{6.6}$$

Eventually, we consider solving a combined polynomial system from Equation (6.5) and (6.6), which is $\{g_1, \dots, g_{2n}\}$. The complexity of computing a Gröbner basis for the ideal

$I = \langle g_1, \dots, g_{2n} \rangle$ generated by polynomial g_1, \dots, g_{2n} is related to its degree of regularity, which is the index of regularity of ideal generated by the homogeneous component of the highest degree of g_1, \dots, g_{2n} , denoted by

$$\tilde{I} = \langle \tilde{g}_1, \dots, \tilde{g}_{2n} \rangle.$$

To analyze this index of regularity, we need to understand the Hilbert function, Hilbert series and Hilbert polynomial of \tilde{I} first.

We consider the quotient ring $\mathbb{F}_q[\mathbf{x}]/\tilde{I}$, which is also a graded ring, namely

$$\mathbb{F}_q[\mathbf{x}]/\tilde{I} = \bigoplus_{d=0}^{\infty} R_d, \quad (6.7)$$

where R_d are homogeneous elements of \tilde{I} of degree d . Then we can define its Hilbert function

$$HF_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}} : d \rightarrow \dim_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}} R_d,$$

which maps an integer d onto the dimension of the \mathbb{F}_q -vector space R_d , and its Hilbert series is the formal series given by

$$HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}}(t) = \sum_{d=0}^{\infty} HF_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}}(d) t^d.$$

Moreover, from Proposition 3.2.4, the Hilbert series of \tilde{I} can be computed. In $\{\tilde{g}_1, \dots, \tilde{g}_{2n}\}$, the first n polynomials are multivariate quadratic homogeneous in variables x_1, \dots, x_n and the last n polynomials are multivariate homogeneous polynomials of degree p in variables x_1, \dots, x_n , therefore from Proposition 3.2.4, we have

$$HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}}(t) = (1 - t^2)^n (1 - t^p)^n HS_{\mathbb{F}_q[\mathbf{x}]}.$$

Finally, we need to compute the Hilbert series of $\mathbb{F}_q[\mathbf{x}]$, which can be derived from following:

$$\begin{aligned} HS_{\mathbb{F}_q[\mathbf{x}]} &= \sum_{d=0}^{\infty} \dim_{\mathbb{F}_q[\mathbf{x}]} \mathbb{F}_q[\mathbf{x}]_d \cdot t^d \\ &= \sum_{d=0}^{\infty} \binom{n+d-1}{d} t^d \\ &= \frac{1}{(1-t)^n}. \end{aligned} \quad (6.8)$$

Therefore, we have the Hilbert series of \tilde{I} given by

$$HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}}(t) = \frac{(1-t^2)^n(1-t^p)^n}{(1-t)^n}. \quad (6.9)$$

Since for overdetermined polynomial systems, the expansion of Equation (6.9) has negative terms, its Hilbert series therefore is defined to be

$$HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}}(t) = \left[\frac{(1-t^2)^n(1-t^p)^n}{(1-t)^n} \right]_+, \quad (6.10)$$

where $[a]_+$ mean truncating a at its first non positive coefficient and the degree of regularity of I is defined to be the degree of the first term of $HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}}(t)$ that is non-positive.

The aforementioned Hilbert series holds for randomly chosen polynomial system, since in PERN, polynomial systems are all chosen as random, this above formula should hold for PERN.

When hybrid approach of specifying k variables before applying Gröbner basis techniques is considered, suppose the polynomial system after specifying k variables become

$$G' = \{g'_1, \dots, g'_n, g_{n+1}, \dots, g_{2n-k}\},$$

and its homogeneous components of the highest degrees be

$$\tilde{G}' = \{\tilde{g}'_1, \dots, \tilde{g}'_n, \tilde{g}_{n+1}, \dots, \tilde{g}_{2n-k}\},$$

then the Hilbert series of the ideal generated by G' , $I' = \langle g \mid g \in G' \rangle$, will change to the following:

$$HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}'}(t) = \left[\frac{(1-t^2)^n(1-t^p)^{n-k}}{(1-t)^{n-k}} \right]_+, \quad (6.11)$$

where $\tilde{I}' = \langle \tilde{g}' \mid \tilde{g}' \in \tilde{G}' \rangle$ and its degree of regularity is given by the degree of the first term of $HS_{\mathbb{F}_q[\mathbf{x}]/\tilde{I}'}(t)$ that is non-positive.

Our experiments on instances under $n = 3, 4, \dots, 15$ with different p confirmed that the degree of regularity of G is consistent with the estimation from Equation (6.11).

Armed with the above analysis, we can obtain a complexity of the hybrid approach of algebraic attack being

$$\min_{0 \leq k \leq n} O \left(p^k \cdot \binom{n-k+d_{reg}-1}{d_{reg}}^\omega \right), \quad (6.12)$$

where d_{reg} is the degree of regularity of G and $2 < \omega \leq 3$ is the linear algebra constant. We will use this attack to estimate the secure parameters for PERN with $\omega = 2$. When a quantum computer is available, with Grover's algorithm, the following complexity can be achieved:

$$\min_{0 \leq k \leq n} O \left(p^{\frac{k}{2}} \cdot \binom{n - k + d_{reg} - 1}{d_{reg}}^{\omega} \right). \quad (6.13)$$

6.3.3 Other attacks

In fact, the constraint MQ problem is also related to the inhomogeneous short integer solution (SIS) problem, attacks that uses lattices are also available, but they require a large complexity and do not contribute in the parameter choice of PERN as algebraic attack does. In the original work where PERN is published [YWT20], an attack against inhomogeneous SIS problem and a lattice-based key recovery attack are considered.

6.4 Secure parameters

Taking the algebraic attack presented in section 6.3, secure parameters for PERN can be deducted. They are given in Table 6.1.

Table 6.1: Secure parameter (n, p, \hat{p}) for PERN

security level	classical attack only	quantum attack
128 bits	(65, 7, 5)	(80, 15, 11)
192 bits	(100, 7, 5)	(122, 15, 9)
256 bits	(135, 7, 5)	(166, 15, 9)

6.5 Conclusion

We recalled the construction of the multivariate encryption scheme called polynomial equations over real numbers (PERN), which, unlike conventional multivariate encryption schemes, uses randomly chosen polynomial maps in its private key. To enable decryption, its plaintext and ciphertext space are manipulated. Its security is related to the constrained MQ problem.

We estimated the security of PERN via algebraic cryptanalysis in this chapter. Especially, we considered the hybrid approach of polynomial solving using Gröbner basis techniques and exhaustive search. The security of PERN was given though complexity of performing algebraic attacks. This complexity was analyzed in this chapter and confirmed by our experiments. Through this algebraic attack, secure parameters of PERN were also given.

PERN is the first multivariate encryption scheme that made use of the constraint MQ problem, which has not been analyzed enough in this field. Both of PERN and the constraint MQ problem should be studied further and we would like to continue researching in this area. Moreover, in the decryption process of PERN, numerical techniques are used to solve non-linear equations, which have room for improvements and it may have some side channel attack concerns.

Chapter 7

On the Weil descent attack against the MQ problem

Among candidates for post-quantum cryptography, multivariate cryptography is known for constructing good signature schemes with short signature length. Its security depends on the hardness of solving a set of multivariate polynomials that are used as public key in a multivariate cryptographic scheme, which is often referred to as the multivariate quadratic problem (MQ problem). In this chapter, we investigate a method for solving the MQ problem, which first transforms a set of multivariate polynomials over a finite field into a new set of multivariate polynomials over its subfield, and then solve it by using algebraic tools like XL, F4 or F5. We investigate the complexity of using such method by analyzing the non-trivial syzygies and the first fall degree of this resulting new polynomial system. Through this, we give a concrete formula for estimating this first fall degree and verify its correctness through some experiments on some small parameters. For a given multivariate quadratic polynomial system f_1, \dots, f_n over a finite field \mathbb{F}_{2^q} , we show the first fall degree of the polynomial system obtained from applying the Weil descent on f_1, \dots, f_n is independent from the choice of q being

$$\min \left\{ d \mid n \binom{n}{d-2} > \binom{n}{d} + \binom{n}{2} \binom{n}{d-4} \right\}.$$

Moreover, we discuss the possible improvements on this method by coupling it with the hybrid approach of exhaustive search and Gröbner basis techniques.

The result in this chapter is based on an unpublished manuscript.

7.1 Introduction

With currently widely used cryptosystems, RSA [RSA78] and ECC [Kob87], being threatened by the development of quantum computers because of Shor's quantum algorithm [Sho97], research on the post-quantum cryptography has become more urgent. NIST [AASA⁺18, AASA⁺20, C JL⁺16] 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. Currently, their project has entered the third round of screening, where multivariate signature scheme Rainbow [DCP⁺20] is chosen as one of the third round finalists and the signature scheme GeMSS [CFMR⁺20] is chosen as an alternative candidate.

Multivariate cryptography is considered as one of the candidates for post-quantum cryptography because of the hardness of solving the multivariate quadratic problem. Multivariate public key cryptosystems use a set of multivariate quadratic polynomials as its public key, hence solving the polynomials in a public key is equivalent to breaking a multivariate cryptosystem. Fukuoka MQ challenge [YDH⁺15a] is dedicated to understanding the hardness of the multivariate quadratic problem.

One of the most efficient way to solve the multivariate quadratic problem is through computing a Gröbner basis [Buc65] of the ideal generated by the given set of polynomials, which is proposed along with an algorithm called Buchberger's algorithm. Later on, new algorithms F4 [Fau99], F5 [Fau02] and XL [CKPS00], which use linear algebra to do polynomial reduction, are proposed. This really changed the security analysis on multivariate cryptography. For example, Gröbner bases are used to break the first HFE [Pat96] challenge [FJ03].

The Weil descent attack [FR94] was first proposed to break the discrete logarithm problem on algebraic curve over composite fields, and it can also be applied to the multivariate quadratic problem. When a set of multivariate quadratic polynomials over a field are given, through the Weil descent, a new polynomial system over its subfield can be obtained. Adding the relation on the new variables in the subfield to this new polynomial system will give us a very large polynomial system. Inverting HFE is, in fact, equivalent to applying the Weil descent attack on a univariate polynomial over a large field and some studies have used it to investigate the security of HFE [DH11, DK12, DY13, DG10, HKYY18, HKY15]. In [CP12, FPPR12, FPPR11], the case of applying

the Weil descent on a high-degree multivariate polynomial over a finite field is discussed. However, it is still unclear whether this Weil descent transformation makes a difference on solving the multivariate quadratic problem and we want to fill in this gap in our work.

7.1.1 Contribution

The main contribution of this chapter is giving a complexity analysis on the Weil descent against the MQ problem. More specifically, we analyze the first fall degree of the polynomial system obtained from the Weil descent against the MQ problem by considering its non-trivial syzygies. For a given polynomial system f_1, \dots, f_n over a finite field \mathbb{F}_{2^q} in variables x_1, \dots, x_n , the Weil descent transforms it into a new polynomial system $f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{n,1}, \dots, f'_{n,q}$ over \mathbb{F}_2 in variables $y_{1,1}, \dots, y_{1,q}, \dots, y_{n,1}, \dots, y_{n,q}$. Since those variables are over \mathbb{F}_2 , they all hold $y_{i,j}^2 - y_{i,j}$ for $i = 1, \dots, n$ and $j = 1, \dots, q$. We find that the first fall degree of $f'_{1,1}, \dots, f'_{n,q}$ is independent from the choice of q and is only determined by n :

$$\min \left\{ d \mid n \binom{n}{d-2} > \binom{n}{d} + \binom{n}{2} \binom{n}{d-4} \right\}.$$

Moreover, we think solving $f'_{1,1}, \dots, f'_{n,q}$ works well with the hybrid approach of exhaustive search and Gröbner basis techniques, since variables $y_{1,1}, \dots, y_{n,q}$ are over \mathbb{F}_2 , which makes them easy to be specified. Specifying every variable cost a complexity of 2. Furthermore, algorithms on solving boolean multivariate polynomials such as the Crossbred [JV18] can also be applied for further improvements.

7.1.2 Organization

This chapter is organized as follows. Section 7.2 explains about the multivariate quadratic (MQ) problem, complexity of solving the MQ problem using Gröbner basis techniques, the degree of regularity, the first fall degree and computing syzygies of a set of polynomials using linear algebra. In section 7.3, we introduce the Weil descent transformation on a set of multivariate quadratic polynomials and its complexity. In section 7.4, we run some experiments on the Weil descent against the MQ problem under some small parameters and we compare the complexity of directly solving the MQ problem using Gröbner basis technique with applying the Weil descent on the MQ problem, and discuss a possible improvement. In section 7.5, we conclude this chapter.

7.2 The multivariate quadratic problem

In this section, we review the multivariate quadratic problem, a mathematical tool used for solving it called Gröbner bases and the complexity for computing a Gröbner basis.

7.2.1 Multivariate quadratic problem

Let \mathbb{F} be a finite field of order q , $m, n \in \mathbb{N}$, and $R := \mathbb{F}[x_1, \dots, x_n]$ be the polynomial ring in variables x_1, \dots, x_n over \mathbb{F} . The multivariate quadratic problem (MQ problem) is defined in Definition 2.1.1 and an effective method for solving this problem is through Gröbner basis computation [Buc65]. Efficient algorithms for computing a Gröbner basis include XL [CKPS00], F4 [Fau99] and F5 [Fau02]. The complexity of using those algorithms for computing a Gröbner basis depends on polynomials involved during this computation, hence the polynomial degree at which an algorithm terminates can be used to estimate this complexity. This degree is often referred to as *solving degree*, denoted by d_{sol} . This complexity mainly comes from a computation of the row echelon form of a Macaulay matrix of degree d_{sol} . Suppose such a Macaulay matrix has size $R_{d_{sol}} \times C_{d_{sol}}$, then the complexity of the fast algorithm proposed in [Sto10] for computing its row echelon form is given by $O(R_{d_{sol}} C_{d_{sol}}^{\omega-1})$, where $2 \leq \omega \leq 3$ is the linear algebra constant. However, the solving degree of a polynomial system is an experimental value, it can be difficult to be precisely known. Therefore, an approximation value, which is related to the mathematical property of the given polynomial system, is often used. It is called *degree of regularity* (d_{reg}) [BFSY05], which is defined in Definition 3.3.15.

The degree of regularity d_{reg} for random systems can be precisely evaluated, but hard to estimate for specific families of polynomial systems. Therefore, in cryptographic studies, d_{reg} is often approximated by the *first fall degree* (d_{ff}), which is defined in Definition 5.2.2.

Many results on multivariate cryptosystems are based on analyzing d_{ff} [DK12, DH11, DY13], although it is not always true that d_{ff} and d_{reg} are very close, experimental and theoretical evidences in these results have shown it seems to be true for some cryptographic schemes.

7.2.2 Computing syzygies using linear algebra

In this subsection, we exploit the method for computing syzygies of a set of homogeneous polynomials using linear algebra.

Given homogeneous polynomials $f_1, \dots, f_m \in R$, its degree 0 syzygies $(s_1^{(0)}, \dots, s_m^{(0)})$ satisfies

$$(f_1, \dots, f_m) \cdot \begin{pmatrix} s_1^{(0)} \\ \vdots \\ s_m^{(0)} \end{pmatrix} = 0.$$

Let \mathbf{m}_0 be the set of all monomials appeared in f_1, \dots, f_m , and $\mathbf{c}_i \in \mathbb{F}^{|\mathbf{m}_0|}$ for $i = 1, \dots, m$ be coefficients of f_i with respect to \mathbf{m}_0 . Then we have

$$\mathbf{m}_0 \cdot (\mathbf{c}_1^\top \quad \mathbf{c}_2^\top \quad \cdots \quad \mathbf{c}_m^\top) \cdot \begin{pmatrix} s_1^{(0)} \\ \vdots \\ s_m^{(0)} \end{pmatrix} = 0.$$

Therefore, $(s_1^{(0)}, \dots, s_m^{(0)})$ can be obtained from the right kernel of the matrix $(\mathbf{c}_1^\top \quad \mathbf{c}_2^\top \quad \cdots \quad \mathbf{c}_m^\top)$.

For degree 1 syzygies $(s_1^{(1)}, \dots, s_m^{(1)})$, we assume $s_i^{(1)} = \sum_{j=1}^n a_{i,j} x_j$ for $a_{i,j} \in \mathbb{F}$. Let \mathbf{m}_1 be the set of all monomials appeared in $x_i f_1, \dots, x_i f_m$ for $i = 1, \dots, n$ and $\mathbf{c}_{k,l} \in \mathbb{F}^{|\mathbf{m}_0|}$ be coefficients of $x_k f_l$ with respect to \mathbf{m}_1 . Then we have

$$\mathbf{m}_1 \cdot (\mathbf{c}_{1,1}^\top \quad \mathbf{c}_{2,1}^\top \quad \cdots \quad \mathbf{c}_{n,m}^\top) \cdot \begin{pmatrix} a_{1,1} \\ a_{1,2} \\ \vdots \\ a_{m,n} \end{pmatrix} = 0.$$

Therefore, (s_1^1, \dots, s_m^1) can be obtained from the right kernel of the matrix $(\mathbf{c}_{1,1}^\top \quad \mathbf{c}_{2,1}^\top \quad \cdots \quad \mathbf{c}_{n,m}^\top)$.

Similarly, for degree d syzygies $(s_1^{(d)}, \dots, s_m^{(d)})$, we first find the set of degree d monomials $\mathbf{b} = (b_1, \dots, b_t)$ in $\mathbb{F}[x_1, \dots, x_n]$. Then consider polynomials $b_i f_j$. Let \mathbf{m}_d be all monomials appeared in $b_i f_j$ and $\mathbf{c}_{k,j} \in \mathbb{F}^{|\mathbf{m}_d|}$ be coefficients of $b_k f_l$. Then $(s_1^{(d)}, \dots, s_m^{(d)})$ can be obtained from the left kernel of the matrix $(\mathbf{c}_{1,1}^\top \quad \mathbf{c}_{2,1}^\top \quad \cdots \quad \mathbf{c}_{t,m}^\top)$.

7.3 Weil descent on the MQ problem

In this section, we describe the Weil descent transformation on the MQ problem, which first transforms a set of multivariate polynomials over a finite field to a new set of multivariate polynomials over its subfields and then solve it using algebraic methods. We especially focus on finite fields with characteristic 2.

7.3.1 Weil descent transformation on the MQ problem

Let \mathbb{F}_{2^q} be a finite field of characteristic 2 with a cardinality 2^q , $\mathbb{F}_{2^q}[x_1, \dots, x_n]$ be the polynomial ring in variables x_1, \dots, x_n over \mathbb{F}_{2^q} . Let $\{\theta_1, \dots, \theta_{\frac{q}{d}}\} \subset \mathbb{F}_{2^q}$ be a basis for $\mathbb{F}_{2^q}/\mathbb{F}_{2^d}$, where $d|q$ holds. Let $\mathbb{F}_{2^d}[y_{1,1}, y_{1,2}, \dots, y_{1,\frac{q}{d}}, \dots, y_{m,\frac{q}{d}}]$ be the polynomial ring in $y_{1,1}, \dots, y_{m,\frac{q}{d}}$ over the finite field \mathbb{F}_{2^d} . Then $x_i = \sum_{j=1}^{\frac{q}{d}} y_{i,j} \theta_j$ holds for $i = 1, \dots, n$.

In the MQ problem, a polynomial system

$$\begin{cases} f_1(x_1, \dots, x_n) = 0 \\ \vdots \\ f_m(x_1, \dots, x_n) = 0 \end{cases}$$

is asked to be solved. If we substitute $\sum_{j=1}^{\frac{q}{d}} y_{i,j} \theta_j$ in this polynomial system for x_i , we obtain a new system given by

$$\begin{cases} f'_{1,1} \theta_1 + \dots + f'_{1,\frac{q}{d}} \theta_{\frac{q}{d}} = 0 \\ \vdots \\ f'_{m,1} \theta_1 + \dots + f'_{m,\frac{q}{d}} \theta_{\frac{q}{d}} = 0 \\ y_{1,1}^{2^d} - y_{1,1} = 0 \\ \vdots \\ y_{n,\frac{q}{d}}^{2^d} - y_{n,\frac{q}{d}} = 0 \end{cases} \Rightarrow \begin{cases} f'_{1,1} = 0 \\ f'_{1,2} = 0 \\ \vdots \\ f'_{1,\frac{q}{d}} = 0 \\ \vdots \\ f'_{m,\frac{q}{d}} = 0 \\ y_{1,1}^{2^d} - y_{1,1} = 0 \\ \vdots \\ y_{n,\frac{q}{d}}^{2^d} - y_{n,\frac{q}{d}} = 0 \end{cases}.$$

Note that equations

$$\begin{cases} y_{1,1}^{2^d} - y_{1,1} = 0 \\ \vdots \\ y_{n,\frac{q}{d}}^{2^d} - y_{n,\frac{q}{d}} = 0 \end{cases}$$

are trivial relations over the field \mathbb{F}_{2^d} and are also commonly referred to as *field equations*.

7.3.2 Complexity analysis

In this subsection, we will investigate the complexity of solving the polynomial system obtained from the Weil descent transformation on a set of multivariate polynomial system through Gröbner bases computation. We will focus on the case of $d = 1$.

When $d = 1$, $f_1, \dots, f_m \in \mathbb{F}_{2^q}[x_1, \dots, x_n]$ transforms into $f'_{1,1}, f'_{1,2}, \dots, f'_{1,q}, f'_{2,1}, \dots, f'_{m,q}, y_{1,1}^2 - y_{1,1}, \dots, y_{n,q}^2 - y_{n,q} \in \mathbb{F}_2[y_{1,1}, \dots, y_{1,q}, y_{2,1}, \dots, y_{n,q}]$, and we want to know the complexity of solving this new polynomial system using Gröbner basis techniques. In the following discussions, we assume $m = n$, and we try to analyze its first fall degree by investigating its syzygies.

Since the first fall degree depends only on the homogeneous components of the highest degree of a polynomial system, we assume polynomials f_1, \dots, f_n are all homogeneous of degree 2 since multivariate quadratic polynomials are used in multivariate cryptography. In this case, we need to analyze the first fall degree of

$$\left\{ \begin{array}{l} f_1(x_1, \dots, x_n) = 0 \\ \vdots \\ f_n(x_1, \dots, x_n) = 0 \\ x_1 - \sum_{j=1}^q y_{1,j} \theta_j = 0 \\ \vdots \\ x_n - \sum_{j=1}^q y_{n,j} \theta_j = 0 \\ y_{1,1}^2 - y_{1,1} = 0 \\ \vdots \\ y_{n,q}^2 - y_{n,q} = 0 \end{array} \right\} = \left\{ \begin{array}{l} f'_{1,1} = 0 \\ \vdots \\ f'_{n,q} = 0 \\ y_{1,1}^2 - y_{1,1} = 0 \\ \vdots \\ y_{n,q}^2 - y_{n,q} = 0 \end{array} \right\}. \quad (7.1)$$

Since f_1, \dots, f_n are all homogeneous quadratic polynomials, $f'_{1,1}, \dots, f'_{n,q}$ are also homogeneous quadratic polynomials. To know the first fall degree of (7.1), we need to only consider its homogeneous components of the highest degree, namely, we need to consider the syzygies of the system

$$\{f'_{1,1} = 0, \dots, f'_{n,q} = 0, y_{1,1}^2 = 0, \dots, y_{n,q}^2 = 0\}. \quad (7.2)$$

The speciality about the system (7.2) is that during a Gröbner basis computation, any

monomials that contain $y_{i,j}^2$ for $i = 1, \dots, n$ and $j = 1, \dots, q$ vanish. Moreover, we know

$$\begin{cases} x_1^2 = \sum_{j=1}^q y_{1,j}^2 \theta_j^2 \\ \vdots \\ x_n^2 = \sum_{j=1}^q y_{n,j}^2 \theta_j^2 \end{cases} \quad (7.3)$$

due to Frobenius endomorphism, which implies any monomials that contain x_i^2 for $i = 1, \dots, n$ vanish during a Gröbner basis computation.

Next we try to analyze the first fall degree of the system (7.2) starting from the case $n = m = 2$, then increase to larger cases.

Case $n = m = 2$

When $n = m = 2$, monomials involved in f_1 and f_2 can only be x_1^2, x_1x_2 and x_2^2 since f_1 and f_2 are homogeneous of quadratic polynomials, and we want to find a Gröbner basis for the ideal generated by f_1, f_2 .

Assume we have

$$\begin{cases} f_1 = a_{1,1}x_1^2 + a_{1,2}x_1x_2 + a_{2,2}x_2^2, \\ f_2 = b_{1,1}x_1^2 + b_{1,2}x_1x_2 + b_{2,2}x_2^2. \end{cases}$$

By extracting the coefficients of f_1 and f_2 , we form a Macaulay matrix

$$\begin{matrix} & x_1^2 & x_1x_2 & x_2^2 \\ f_1 & \begin{pmatrix} a_{1,1} & a_{1,2} & a_{2,2} \end{pmatrix} \\ f_2 & \begin{pmatrix} b_{1,1} & b_{1,2} & b_{2,2} \end{pmatrix} \end{matrix}.$$

Performing row operations on this matrix is equivalent to doing operations on f_1 and f_2 . Now if we do not consider $y_{1,1}, \dots, y_{2,q}$, the echelon form of this matrix remains the same shape. However, if $y_{1,1}^2 = 0, \dots, y_{2,q}^2 = 0, x_1^2 = 0, x_2^2 = 0$ are considered, entries $a_{1,1}, a_{2,2}, b_{1,1}, b_{2,2}$ vanish and this matrix turns into

$$\begin{matrix} & x_1^2 & x_1x_2 & x_2^2 \\ f_1 & \begin{pmatrix} 0 & a_{1,2} & 0 \end{pmatrix} \\ f_2 & \begin{pmatrix} 0 & b_{1,2} & 0 \end{pmatrix} \end{matrix},$$

and associated polynomials are $\hat{f}_1 = a_{1,2}x_1x_2$ and $\hat{f}_2 = b_{1,2}x_1x_2$. Since \hat{f}_1 and \hat{f}_2 are scalar times to each other, when the Weil descent transformation are applied on them, among the obtained new polynomials, some polynomials are scalar times to each other. Assume applying the Weil descent on \hat{f}_1 and \hat{f}_2 gives us $f'_{1,1}, \dots, f'_{1,q}, f'_{2,1}, \dots, f'_{2,q}$ and

$y_{1,1}^2, \dots, y_{2,q}^2$. We know $f'_{1,1}, \dots, f'_{1,q}$ and $f'_{2,1}, \dots, f'_{2,q}$ are scalar times to each other. There exist degree 0 syzygies of the system $f'_{1,1}, \dots, f'_{2,q}, y_{1,1}^2, \dots, y_{2,q}^2$ since the right kernel of the coefficient matrix of this polynomial system has dimension q , and they are not trivial syzygies since their degree is 0. Hence the first fall degree of this polynomial system is 2 regardless of the choice of q .

Case $n = m = 3$

When $n = m = 3$, f_1, f_2, f_3 can only possibly have monomials $x_1^2, x_1x_2, x_1x_3, x_2^2, x_2x_3, x_3^2$. We can assume polynomial f_1, f_2, f_3 are

$$\begin{cases} f_1 = a_{1,1}x_1^2 + a_{1,2}x_1x_2 + a_{1,3}x_1x_3 + a_{2,2}x_2^2 + a_{2,3}x_2x_3 + a_{3,3}x_3^2, \\ f_2 = b_{1,1}x_1^2 + b_{1,2}x_1x_2 + b_{1,3}x_1x_3 + b_{2,2}x_2^2 + b_{2,3}x_2x_3 + b_{3,3}x_3^2, \\ f_3 = c_{1,1}x_1^2 + c_{1,2}x_1x_2 + c_{1,3}x_1x_3 + c_{2,2}x_2^2 + c_{2,3}x_2x_3 + c_{3,3}x_3^2, \end{cases}$$

where $a_{i,j}, b_{i,j}, c_{i,j} \in \mathbb{F}_{2^q}$. Its corresponding Macaulay matrix is

$$\begin{matrix} & x_1^2 & x_1x_2 & x_1x_3 & x_2^2 & x_2x_3 & x_3^2 \\ \begin{matrix} f_1 \\ f_2 \\ f_3 \end{matrix} & \begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{2,2} & a_{2,3} & a_{3,3} \\ b_{1,1} & b_{1,2} & b_{1,3} & b_{2,2} & b_{2,3} & b_{3,3} \\ c_{1,1} & c_{1,2} & c_{1,3} & c_{2,2} & c_{2,3} & c_{3,3} \end{pmatrix} \end{matrix}.$$

When considering $y_{1,1}^2 = 0, \dots, y_{3,q}^2 = 0$, this matrix reduces to

$$\begin{matrix} & x_1^2 & x_1x_2 & x_1x_3 & x_2^2 & x_2x_3 & x_3^2 \\ \begin{matrix} \bar{f}_1 \\ \bar{f}_2 \\ \bar{f}_3 \end{matrix} & \begin{pmatrix} 0 & a_{1,2} & a_{1,3} & 0 & a_{2,3} & 0 \\ 0 & b_{1,2} & b_{1,3} & 0 & b_{2,3} & 0 \\ 0 & c_{1,2} & c_{1,3} & 0 & c_{2,3} & 0 \end{pmatrix} \end{matrix}.$$

Performing Gaussian Elimination on this matrix gives us

$$\begin{matrix} & x_1^2 & x_1x_2 & x_1x_3 & x_2^2 & x_2x_3 & x_3^2 \\ \begin{matrix} \hat{f}_1 \\ \hat{f}_2 \\ \hat{f}_3 \end{matrix} & \begin{pmatrix} 0 & \hat{a}_{1,2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \hat{b}_{1,3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \hat{c}_{2,3} & 0 \end{pmatrix} \end{matrix}.$$

The ideal generated by f_1, f_2, f_3 is the same as the ideal generated by $\hat{f}_1 = \hat{a}_{1,2}x_1x_2, \hat{f}_2 = \hat{b}_{1,3}x_1x_3, \hat{f}_3 = \hat{c}_{2,3}x_2x_3$.

Let $f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{3,1}, \dots, f'_{3,q}, y_{1,1}^2, \dots, y_{3,q}^2 - y_{3,q}$ be the new polynomial system obtained after the Weil descent transformation. For degree 0 syzygies of $\{f'_{1,1}, \dots, f'_{3,1}, y_{1,1}^2, \dots, y_{3,1}^2\}$ to exist, there need to exist some dependent relations between polynomials $\hat{f}_1, \hat{f}_2, \hat{f}_3$, which can only happen when any of

$\hat{a}_{1,2}, \hat{b}_{1,3}, \hat{c}_{2,3}$ is 0. This happens with probability $\frac{1}{q}$. Therefore, the first fall degree of $\{f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{3,1}, \dots, f'_{3,q}, y_{1,1}^2 - y_{1,1}, \dots, y_{3,q}^2 - y_{3,q}\}$ with probability $\frac{1}{q}$ is 2.

Before considering degree 1 syzygies, we first perform the Weil descent transformation on $\hat{f}_1, \hat{f}_2, \hat{f}_3$ with

$$\begin{cases} x_1 = \sum_{j=1}^q y_{1,j} \theta_j, \\ x_2 = \sum_{j=1}^q y_{2,j} \theta_j, \\ x_3 = \sum_{j=1}^q y_{3,j} \theta_j. \end{cases}$$

Let $\Delta = \begin{pmatrix} \theta_1 & \dots & \theta_q \end{pmatrix}^\top \cdot \begin{pmatrix} \theta_1 & \dots & \theta_q \end{pmatrix}$, we have

$$\begin{cases} \hat{f}_1 = \hat{a}_{1,2} \cdot (y_{1,1}, \dots, y_{1,q}) \cdot \Delta \cdot (y_{2,1}, \dots, y_{2,q}), \\ \hat{f}_2 = \hat{b}_{1,3} \cdot (y_{1,1}, \dots, y_{1,q}) \cdot \Delta \cdot (y_{3,1}, \dots, y_{3,q}), \\ \hat{f}_3 = \hat{c}_{2,3} \cdot (y_{2,1}, \dots, y_{2,q}) \cdot \Delta \cdot (y_{3,1}, \dots, y_{3,q}). \end{cases}$$

To find the degree 1 syzygies of $\{\hat{f}_1, \hat{f}_2, \hat{f}_3\}$, we only need to find the right kernel of the coefficient matrix of $\{y_{1,1}\hat{f}_1, \dots, y_{3,q}\hat{f}_1, y_{1,1}\hat{f}_2, \dots, y_{3,q}\hat{f}_2, y_{1,1}\hat{f}_3, \dots, y_{3,q}\hat{f}_3\}$. We also need to consider the fact that all monomials that contain $y_{1,1}^2, \dots, y_{3,q}^2$ vanish. However, it seems like a very difficult task and we consider another path for this problem.

As we have discussed before, we showed x_1^2, x_2^2, x_3^2 all vanish due to $y_{1,1}^2, \dots, y_{3,q}^2$ (see (7.3)), which can be easily obtained from Frobenius endomorphism. Therefore, instead of multiplying $y_{1,1}, \dots, y_{3,q}$ with $\hat{f}_1, \hat{f}_2, \hat{f}_3$, we multiply x_1, x_2, x_3 with $\hat{f}_1, \hat{f}_2, \hat{f}_3$. We obtain

$$\begin{cases} g_1 = x_1 \hat{f}_1 = \hat{a}_{1,2} x_1^2 x_2 = 0, \\ g_2 = x_2 \hat{f}_1 = \hat{a}_{1,2} x_1 x_2^2 = 0, \\ g_3 = x_3 \hat{f}_1 = \hat{a}_{1,2} x_1 x_2 x_3, \\ g_4 = x_1 \hat{f}_2 = \hat{b}_{1,3} x_1^2 x_3 = 0, \\ g_5 = x_2 \hat{f}_2 = \hat{b}_{1,3} x_1 x_2 x_3, \\ g_6 = x_3 \hat{f}_2 = \hat{b}_{1,3} x_1 x_3^2 = 0, \\ g_7 = x_1 \hat{f}_3 = \hat{c}_{2,3} x_1 x_2 x_3, \\ g_8 = x_2 \hat{f}_3 = \hat{c}_{2,3} x_2^2 x_3 = 0, \\ g_9 = x_3 \hat{f}_3 = \hat{c}_{2,3} x_2 x_3^2 = 0. \end{cases}$$

Now we observe g_1, \dots, g_9 . Take $g_1 = x_1 \hat{f}_1 = 0$ for example. Since $x_1 = \sum_{j=1}^q y_{1,j} \theta_j$, we have

$$g_1 = \theta_1 y_{1,1} \hat{f}_1 + \dots + \theta_q y_{1,q} \hat{f}_1 = 0,$$

which means $(\sum_{j=1}^q \theta_j y_{1,j}, 0, 0)$ is a degree 1 syzygy of $\{\hat{f}_1, \hat{f}_2, \hat{f}_3\}$. Similarly, from $g_2, g_4, g_6, g_7, g_8, g_9$ we can also obtain degree 1 syzygies. Moreover, we have

$$\begin{cases} g_3 &= x_3 \hat{f}_1 = \hat{a}_{1,2} x_1 x_2 x_3, \\ g_5 &= x_2 \hat{f}_2 = \hat{b}_{1,3} x_1 x_2 x_3, \\ g_7 &= x_1 \hat{f}_3 = \hat{c}_{2,3} x_1 x_2 x_3, \end{cases}$$

which are linearly dependent to each other. Therefore, g_5 and g_7 can also be reduced to 0 by g_3 , which give two more syzygies. Therefore, in total we get 8 degree 1 syzygies of $\hat{f}_1, \hat{f}_2, \hat{f}_3$. Since trivial syzygies all have degree 2, these syzygies can only be non-trivial syzygies and the first fall degree of $\{f'_{1,1}, \dots, f'_{3,q}, y_{1,1}^2 - y_{1,1}, \dots, y_{3,q}^2 - y_{3,q}\}$ is 3 with a probability of $1 - \frac{1}{q}$.

Case $n = m > 3$

Armed with results from case $n = m = 2$ and $n = m = 3$, we generalize those results to higher parameter cases.

Consider homogeneous quadratic polynomials f_1, \dots, f_n . After the Weil descent transformation, suppose we have polynomials

$$f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{n,1}, \dots, f'_{n,q}, \quad (7.4)$$

$$y_{1,1}^2 - y_{1,1}, \dots, y_{n,q}^2 - y_{n,q}. \quad (7.5)$$

Because of (7.5), we have $x_i^2 = 0$ for $i = 1, \dots, n$. Therefore, monomials x_i^2 involved in f_1, \dots, f_n will vanish after the Weil descent transformation, which means total number of monomials remained are $\binom{n}{2}$.

For degree 0 syzygies, polynomials f_1, \dots, f_n have to be linearly dependent, which is highly improbable. As for syzygies with degree d , we simply multiply monomials of degree d in variables x_1, \dots, x_n with polynomials f_1, \dots, f_n and observe its coefficient matrix.

For $n = m = 3$, if we consider degree 1 syzygies, we multiply x_1, \dots, x_n with f_1, \dots, f_n which gives us n^2 new polynomials. To get its coefficient matrix, we need to know the total number of monomials involved. Among new polynomials, monomials can not contain x_1^2, \dots, x_n^2 as they will vanish. Therefore, we will have $\binom{n}{3}$ monomials. Therefore, this coefficient matrix is a $n^2 \times \binom{n}{3} = 9 \times 1$ matrix, which has a dimension 8 kernel space, and this is consistent with our results in $n = m = 3$ case.

For $n = m > 3$, if we consider degree d syzygies, we multiply monomials of degree d with square free in variables x_1, \dots, x_n with f_1, \dots, f_n , which results in $n \cdot \binom{n}{d}$ new polynomials of degree $d + 2$. To know about the size of the coefficient matrix of those new polynomials, we need to count the total number of monomials involved. Considering they are all degree $d + 2$ polynomials with square free monomials, the total number of monomials should be $\binom{n}{d+2}$. Therefore, its coefficient matrix should be in size $n\binom{n}{d} \times \binom{n}{d+2}$. When

$$n\binom{n}{d} > \binom{n}{d+2},$$

this matrix will have a dimension $n\binom{n}{d} - \binom{n}{d+2}$ nonzero kernel space. This corresponds to $n\binom{n}{d} - \binom{n}{d+2}$ syzygies, which includes trivial and non-trivial syzygies of

$$f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{n,1}, \dots, f'_{n,q},$$

$$y_{1,1}^2 - y_{1,1}, \dots, y_{n,q}^2 - y_{n,q}.$$

For $d \geq 2$, there exists $\binom{n}{2}\binom{n}{d-2}$ independent trivial syzygies. Therefore, for $d \geq 2$, when $n\binom{n}{d} > \binom{n}{d+2} + \binom{n}{2}\binom{n}{d-2}$ holds, the first fall degree is $d + 2$.

From the aforementioned discussions, we know the first fall degree of $f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{n,1}, \dots, f'_{n,q}, y_{1,1}^2 - y_{1,1}, \dots, y_{n,q}^2 - y_{n,q}$ is independent from the choice of q and only depends on n . And for different choice of n , its d_{ff} is listed in Table 7.1.

Table 7.1: Estimated first fall degree of the polynomial system f_1, \dots, f_n over \mathbb{F}_{2^q} after the Weil descent transformation to polynomials over \mathbb{F}_2

n	$d_{ff}(f_1, \dots, f_n)$ after the Weil descent
2	2
3, ..., 8	3
9, ..., 15	4
16, ..., 23	5
24, ..., 31	6
32, ..., 39	7
40, ..., 46	8
69, ..., 86	9

Remark 7.3.1. When the given polynomial system is a system of m polynomials in n variables over \mathbb{F}_2^q . The first fall degree of the resulting polynomial system from the Weil Descent transformation is

$$\min \left\{ d \mid m \binom{n}{d-2} > \binom{n}{d} + \binom{m}{2} \binom{n}{d-4} \right\}.$$

Moreover, throughout the chapter, the finite fields we used are all fields of characteristic 2, but our results can be easily extended to other fields of different characteristic.

7.4 Experiments and comparison

In this section, we first verify the results obtained from last section by performing some experiments on different polynomial system using the Weil descent transformation. Then evaluate the effect of the Weil descent on f_1, \dots, f_n by comparing it with solving directly using Gröbner basis.

7.4.1 Experiments

We run some experiments on the actual first fall degree and solving degree of f_1, \dots, f_n after the Weil descent transformation. The results are listed in Table 7.2. From these results, our theoretical estimation on the first fall degree of $\{f'_{1,1}, \dots, f'_{1,q}, \dots, f'_{n,q}, y_{1,1}^2 - y_{1,1}, \dots, y_{n,q}^2 - y_{n,q}\}$ is verified. However, there are cases when the solving degree is larger than the first fall degree such as $n = 7, 8$ case.

7.4.2 Comparison with direct solving

Since when the Weil descent transformation is applied to a set of polynomial system f_1, \dots, f_n over a finite field \mathbb{F}_{2^q} and results in a new polynomial system $f'_{1,1}, \dots, f'_{1,q} - 1, q, \dots, f'_{n,1}, \dots, f'_{n,q}$ over the finite field \mathbb{F}_2 , new informations

$$\begin{cases} y_{1,1}^2 - y_{1,1}, \\ \vdots \\ y_{n,q}^2 - y_{n,q} \end{cases}$$

can be added, and it may change the behavior of computing a Gröbner basis. In this subsection, we compare the complexity between with the Weil descent transformation

Table 7.2: Experimented first fall degree and solving degree of the polynomial system f_1, \dots, f_n over \mathbb{F}_{2^q} after the Weil descent Transformation to polynomials over \mathbb{F}_2

n	$q = 3$		$q = 4$		$q = 5$		$q = 6$		$q = 7$	
	d_{ff}	d_{sol}	d_{ff}	d_{sol}	d_{ff}	d_{sol}	d_{ff}	d_{sol}	d_{ff}	d_{sol}
2	2/3	3	2/3	3	2/3	3	2/3	3	2/3	3
3	3	3	3	3	3	3	3	3	3	3
4	3	3	3	3	3	3	3	3	3	3
5	3	3	3	3	3	3	3	3	3	3
6	3	3	3	3	3	3	3	3	3	3
7	3	4	3	4	3	4	3	4	3	4
8	3	4	3	4	3	4	3	4	3	4
9	4	4	4	4	4	4	4	4	4	4
10	4	4	4	4	4	4	4	4	4	4
11	4	4	4	4	4	4	4	4	4	4
12	4	5	4	5	4	5	4	5	4	5
13	4	5	4	5	4	5	4	5	4	5

and without it.

For a given random polynomial system f_1, \dots, f_n in variables x_1, \dots, x_n over a finite field \mathbb{F}_{2^q} , it is supposed to be regular, and its degree of regularity is supposed to $n + 1$ [BFSY05]. The complexity for computing a Gröbner basis using F4/F5 algorithm [Fau99, Fau02] is estimated to be [BFS15]

$$\left(\binom{n + d_{reg}}{d_{reg}} \right)^\omega, \quad (7.6)$$

where d_{reg} is degree of regularity and $2 \leq \omega \leq 3$ is the linear algebra constant.

As for the polynomial system obtained from applying the Weil descent transformation on f_1, \dots, f_n , we know it is not random, and its degree of regularity can not be estimated using the results in [BFSY05], we use its first fall degree as an approximation to its degree of regularity. The complexity hence is

$$\left(\binom{nq + d_{ff}}{d_{ff}} \right)^\omega, \quad (7.7)$$

where d_{ff} is the first fall degree.

Both of (7.6) and (7.7) mean the approximated number of field operations, since (7.6) is for \mathbb{F}_{2^q} and (7.7) is for \mathbb{F}_2 . Suppose an operation over \mathbb{F}_{2^q} is equivalent to q^2 operations over \mathbb{F}_2 , we then have to multiply (7.6) with q^2 .

Take $n = 11$ for example, assume $\omega = 2.8$, then direct solving requires a complexity of

$$\binom{11+12}{12}^{2.8} \cdot q^2 \approx 2^{57.0+2lg(q)},$$

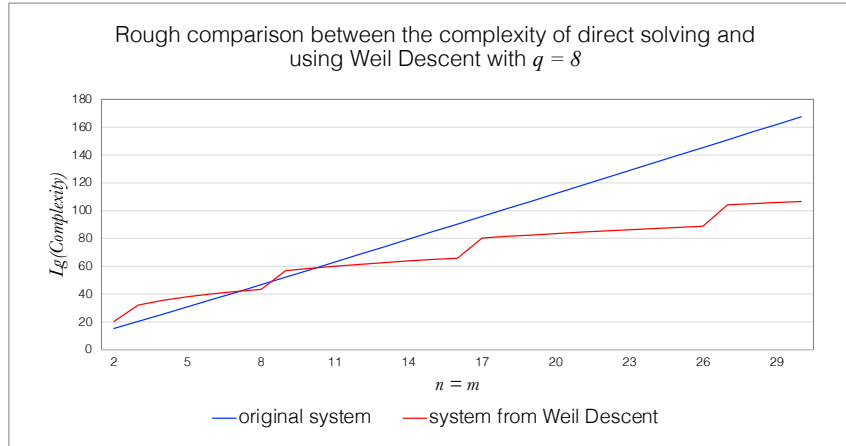
which are $2^{59.03}, 2^{60.20}, 2^{61.03}, 2^{61.67}, 2^{62.20}, 2^{62.64}, 2^{63.03}$ for $q = 2, 3, 4, 5, 6, 7, 8$. For the system derived from the Weil descent, we have a complexity of

$$\binom{11q+4}{4}^{2.8},$$

which are $2^{38.83}, 2^{44.83}, 2^{49.20}, 2^{52.63}, 2^{55.46}, 2^{57.86}, 2^{60.00}$ for $q = 2, \dots, 8$. In this case, the polynomial system from the Weil descent requires less complexity than direct solving.

To give a rough comparison between the complexity of the direct solving and using the Weil descent, we fix $q = 8$ and compare their complexity with different values of n from 2 to 30 using d_{ff} we estimated in Table 7.1 and plot the results in Figure 7.1.

Figure 7.1: Comparison between the complexity of directly solving f_1, \dots, f_n over \mathbb{F}_{2^8} using Gröbner basis techniques with the complexity of solving a new polynomial system over \mathbb{F}_2 obtained from applying the Weil descent transformation on f_1, \dots, f_n



7.4.3 Further improvement with the hybrid approach

When solving f_1, \dots, f_n over a finite field \mathbb{F}_{2^q} with large q with Gröbner basis techniques, applying the hybrid approach [BFP09] does not bring any positive results, as specifying every variable in \mathbb{F}_{2^q} requires a complexity of 2^q . However, when the Weil descent transformation is applied to f_1, \dots, f_n , a new polynomial system

$$\begin{cases} f'_{1,1}(y_{1,1}, \dots, y_{1,q}, \dots, y_{n,1}, \dots, y_{n,q}), \\ \vdots \\ f'_{1,q}(y_{1,1}, \dots, y_{1,q}, \dots, y_{n,1}, \dots, y_{n,q}), \\ \vdots \\ f'_{n,1}(y_{1,1}, \dots, y_{1,q}, \dots, y_{n,1}, \dots, y_{n,q}), \\ \vdots \\ f'_{n,q}(y_{1,1}, \dots, y_{1,q}, \dots, y_{n,1}, \dots, y_{n,q}), \\ y_{1,1}^2 - y_{1,1}, \\ \vdots \\ y_{n,q}^2 - y_{n,q} \end{cases}$$

over \mathbb{F}_2 can be obtained. All new variables are over \mathbb{F}_2 , which means specifying every variable only requires a complexity of 2. Moreover, if we observe Figure 7.1, the complexity for systems from the Weil descent grows like a staircase, not as flat as the original system. This can possibly be fixed by coupling the hybrid approach of polynomial solving with the Weil descent transformation. Further analysis on the behavior of the first fall degree of this approach is left for a future work.

Remark 7.4.1. When a polynomial system over the field \mathbb{F}_2 is to be solved using Gröbner basis techniques, the Macaulay matrices appeared during the process are extremely sparse, which means instead of using Gaussian elimination for computing echelon forms of those matrices, linear algebra techniques for sparse matrices such as block Wiedemann algorithm [Tho02] can be used. In this case, the complexity for solving the polynomial system obtained from the Weil descent is

$$O\left(\binom{nq + d_{ff}}{d_{ff}}^2 \binom{nq}{2}\right).$$

Moreover, for solving polynomial systems over \mathbb{F}_2 , a crossbred algorithm [JV18] is available, and it can be applied to solving the polynomial system derived from the Weil

descent transformation and further improve the performance.

Remark 7.4.2. Among the cryptosystems that are chosen as finalists and candidates in the third round of NIST post-quantum standardization project, there is GeMSS [CFMR⁺17], which is basically a reparameterization of HFE [Pat96] with vinegar and minus modifiers. Both of HFE and GeMSS use a univariate polynomial over a large field as their central map. By applying an isomorphism between this large field and its prime field, this univariate polynomial can be changed into a set of multivariate polynomial over the prime field, which coincides with the rationale of Weil Descent transformation. Therefore, we believe our work can provide some theoretical insights in the cryptanalysis of GeMSS.

7.5 Conclusion

In this chapter, we investigated a method of polynomial solving through the Weil descent transformation on a polynomial system. This method first transforms a polynomial system over a finite field into a new polynomial system over its subfield. The resulting polynomial system ends up having more variables and equations, but trivial relations on those variables can be added to this new polynomial system, which possibly can be solved faster.

We specifically investigated the complexity of this method by theoretically analyzing the syzygies and the first fall degree of the system obtained from the Weil descent. We gave a concrete formula for its first fall degree, and verified our analysis through some experiments on small parameters. Since our analysis are purely theoretical, our results should hold for large parameters as well. However, we acknowledge the complexity of polynomial solving using Gröbner basis computing algorithms are mainly determined by the solving degree rather than the first fall degree. Therefore, the actual solving degree for large parameters may be slightly higher than its first fall degree, and more research should be conducted on this regard. Nevertheless, we think this method of polynomial solving is better than simple direct solving, and it may bring a threat to current multivariate cryptography.

Moreover, we believe when the Weil descent approach couples with the hybrid approach of polynomial solving and the crossbred algorithm, even better results can possibly be obtained and we plan to do more investigations on this.

Chapter 8

Conclusion and future work

8.1 Conclusion

In this thesis, algebraic cryptanalysis on multivariate public-key cryptography is considered, which includes three directions: cryptanalysis of multivariate cryptosystems via existing algebraic techniques, algebraic cryptanalysis of multivariate cryptosystems with structural transformation, and new techniques on polynomials solving.

Algebraic cryptanalysis on multivariate cryptosystems In this direction, this thesis completes algebraic cryptanalysis of multivariate encryption scheme EFC [SDP16] and PERN [YWT20]. EFC is an existing multivariate encryption scheme, whose security depends mainly on the complexity of the algebraic attack, which can be obtained if its degree of regularity is known. Due to its similar construction to HFE [Pat96], its degree of regularity is given using results from HFE type cryptosystems. Different from HFE type cryptosystems, the finite field used in the EFC is \mathbb{F}_2 , which can be used in the algebraic attack since *field equations* and *hybrid approach* can both be effectively applied. This line of research implies the importance of considering algebraic solving techniques such as *field equations*, *hybrid approach* and other techniques. On the other hand, this thesis also gives an algebraic cryptanalysis on multivariate encryption PERN, which is short for polynomial equations over real numbers. It is based on the constraint MQ problem and a generalization of the pq method proposed in [Yas18]. Different from other multivariate cryptosystems, it uses two random multivariate polynomial maps, which can be inverted using numerical techniques, to construct an easy-to-invert

polynomial map. This polynomial map has very small domain of integers and its codomain is a large finite field, this choice is to make this map injective. Due to the small cardinality of its domain, some trivial relations can be used when applying the algebraic attack. In this thesis, those trivial relations are considered when applying the algebraic attack on PERN, and an estimation on its degree of regularity is also given. This line of research implies there exists many variations on the algebraic attack against the multivariate cryptosystems.

Algebraic cryptanalysis with structural transformation In this direction, we specifically focus on algebraic methods for solving the minrank problem and applications in multivariate cryptosystems. The minrank problem related attack called the minrank attack in multivariate public-key cryptography has always been a very powerful tool for breaking cryptosystems. With the research results given in this thesis, we know there are still many research topics left in algebraic cryptanalysis of multivariate cryptosystems with structural transformation. Our research brings more variations for solving the minrank problem. Nevertheless, the asymptotic complexity of solving the minrank problem derived from multivariate public-key cryptosystems has still not been thoroughly studied yet. What's more, there exists other mathematical models that can transform solving the minrank problem into the problem of polynomial solving, these models may bring better complexities compared with existing methods. For example, recently in [War20], new improved minrank successfully broke proposed Rainbow [DS05] parameters, this new attack uses a different mathematical than existing models. This direction of research in multivariate public-key cryptography is a challenging yet very promising topic.

Other techniques on polynomials solving In this direction, complexity analysis of the Weil descent attack on the MQ problem is given. The research presented in this thesis approximates the degree of regularity using the first fall degree, which is related to the non-trivial syzygies of a polynomial system. By analyzing non-trivial syzygies of the system obtained from applying the Weil descent attack on the MQ problem, we could theoretically estimate its first fall degree. Practically, when proposing a cryptosystem, by considering this first fall degree, strongly secure parameters can be obtained.

8.2 Future work

Following the complete research in this thesis, there are some unsolved problems as follows.

Degree of regularity of EFC The degree of regularity of HFE type cryptosystems are derived from approximating their degree of regularity with their first fall degree and analyzing their non-trivial syzygies. HFE type cryptosystems all use univariate polynomials over an extension field and use linear map isomorphism between this extension field and a vector space, it obtains a set of quadratic polynomials. This is very similar to the Weil descent attack on the MQ problem. Following this line of research, it is possible to figure out the first fall degree of EFC with *field equation* and *hybrid approach*.

Degree of regularity related to Weil descent In this thesis, the first fall degree related to Weil descent is given, but the degree of regularity is missing. Analysis in this thesis all use homogeneous components of the highest degree of a polynomial system, there is also another way of obtain homogeneous polynomial systems, which is called homogenization.

Definition 8.2.1 (Homogenization). Let $A = \mathbb{K}[x_1, \dots, x_n]$, $A_h = \mathbb{K}[h, x_1, \dots, x_n]$. The *homogenization morphism* is defined as

$$\begin{aligned} \bullet^h : A &\rightarrow A_h \\ f &\mapsto h^d \cdot f\left(\frac{x_1}{h}, \dots, \frac{x_n}{h}\right), \quad d \text{ is the degree of } f. \end{aligned}$$

Definition 8.2.2 (Dehomogenization).

$$\begin{aligned} \bullet^a : A_h &\rightarrow A \\ f &\mapsto f(H = 1). \end{aligned}$$

Different analysis results can be obtained using homogenized polynomial system, and we need to compare it with the results in chapter 7.

Research related to PERN PERN [YWT20] is recently proposed and its security is not well understood yet. Moreover, numerical techniques are used in its decryption

process, which is not as efficient compared to the existing multivariate encryption schemes. Therefore, its cryptanalysis and efficiency improvement are still left for further research.

Asymptotic complexity of minrank As another powerful attack that broke many multivariate cryptosystems, its asymptotic complexity is still not yet understood. Recently developments in [VBC⁺19, War20, BBB⁺20, BBC⁺20] suggest the minrank attack is more powerful than previous analyzed. Understanding its asymptotic complexity significantly helps understanding the security of multivariate public-key cryptography.

Acknowledgements

The results presented in this thesis could not possibly be obtained without the help of numerous people and I am more than grateful for their support.

First of all, I want to thank my supervisor, Professor Tsuyoshi Takagi, who has been guiding me through my entire graduate study since I first came to Japan. Not only did he introduce me into the world of research, he also offered tremendous amount of support on my research and my living in this foreign country. I especially am grateful for his funding support whenever I needed, which allowed me to conduct research without any worries. He showed me the possibilities in the field of research and forever influenced my trajectory of life. I am really glad to be his student.

I want to thank Dr. Yasuhiko Ikematsu, Dr. Albrecht Petzoldt and Dr. Shuhei Nakamura, who spent a lot of time helping me on my research. Albrecht first lead me into the world of multivariate public-key cryptography and taught me many necessary implementation techniques. Yasuhiko showed me the beauty of mathematics and helped me in every aspect of conducting my research. I feel his influence every time I run experiments and write an academic paper. Finally, I thank Shuhei for the useful discussions we had off campus.

I thank Professor Ludovic Perret and Professor Jean-Charles Faugère for taking me in as a mobility student in Laboratoire d'Informatique de Paris 6 for 3 months, where I learnt many things on polynomial solving and Gröbner bases.

I thank Professor Takanori Yasuda for collaborating on his paper [YWT20], Hiroki Furue for collaborating on his paper [FKI⁺20], Jiahui Chen for collaborating on his paper [CNL⁺20].

Further, I want to thank my friend and former colleague Dr. Yuntao Wang and Weiyao Wang for the activities we had outside of laboratory. I also thank my colleagues in the laboratory and our secretary Kaori Omura for helping me through all the paper work

whenever I needed.

I also thank Japan Society for the Promotion of Science (JSPS) for offering me a fellowship and KAKENHI Grant Number 18J20866 for supporting me on attending different conferences.

Finally, I most importantly want to thank my parents and sisters for all the support and love I received from them while pursuing my study, and I apologize for not visiting them more often. I am forever grateful.

Bibliography

- [AASA⁺18] 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. <https://www.nist.gov/publications/status-report-first-round-nist-post-quantum-cryptography-standardization-process>.
- [AASA⁺20] 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 second round of the nist post-quantum cryptography standardization process. NIST Internal Report 8309, National Institute of Standards and Technology, 2020. <https://csrc.nist.gov/publications/detail/nistir/8309/final>.
- [AFI⁺04] Gwénolé Ars, Jean-Charles Faugère, Hideki Imai, Mitsuru Kawazoe, and Makoto Sugita. Comparison between xl and gröbner basis algorithms. In *Advances in Cryptology – ASIACRYPT 2004*, volume 3329 of *LNCS*, pages 338–353. Springer, 2004. https://doi.org/10.1007/978-3-540-30539-2_24.
- [Ajt96] Miklós Ajtai. Generating hard instances of lattice problems (extended abstract). In *Proceedings of the twenty-eighth annual ACM symposium on Theory of Computing – STOC’96*, volume 1996 of *STOC*, pages 99–108, 1996. <https://doi.org/10.1145/237814.237838>.
- [AM69] Michael F. Atiyah and Ian G. Macdonald. *Introduction To Commutative Algebra*. Addison-Wesley Series in Mathematics. Addison-Wesley Publishing

- Company, 1969. <http://math.univ-lyon1.fr/~mathieu/CoursM2-2020/AMD-ComAlg.pdf>.
- [Ani84] David J. Anick. Thin algebras of embedding dimension three. *Journal of Algebra*, 100(1):235–259, 1984. [https://doi.org/10.1016/0021-8693\(86\)90076-1](https://doi.org/10.1016/0021-8693(86)90076-1).
- [BBB⁺20] Magali Bardet, Pierre Briaud, Maxime Bros, Philippe Gaborit, Vincent Neiger, Olivier Ruatta, and Jean-Pierre Tillich. An algebraic attack on rank metric code-based cryptosystems. In *Advances in Cryptology – EUROCRYPT 2020*, volume 12107 of *LNCS*, pages 64–93. Springer, 2020. https://doi.org/10.1007/978-3-030-45727-3_3.
- [BBC⁺20] Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, and Javier Verbel. Improvements of algebraic attacks for solving the rank decoding and minrank problems. *Computing Research Repository, Cryptography and Security*(2002.08322v3):1–26, 2020. <https://arxiv.org/abs/2002.08322v3>.
- [BCP97] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. <https://doi.org/10.1006/jSCO.1996.0125>.
- [Ber67] E. R. Berlekamp. Factoring polynomials over finite fields. *The Bell System Technical Journal*, 46(8):1853–1859, 1967. <https://doi.org/10.1002/j.1538-7305.1967.tb03174.x>.
- [BFP09] 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. <https://doi.org/10.1515/JMC.2009.009>.
- [BFP12] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret. Solving polynomial systems over finite fields: improved analysis of the hybrid approach. In *ISSAC '12: Proceedings of the 37th International Symposium on Symbolic and Algebraic Computation*, pages 67–74. ACM, 2012. <https://doi.org/10.1145/2442829.2442843>.
- [BFP13] 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. <https://doi.org/10.1007/s10623->

- 012-9617-2.
- [BFS99] 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. <https://doi.org/10.1006/jcss.1998.1608>.
 - [BFS15] Magali Bardet, Jean-Charles Faugère, and Bruno Salvy. On the complexity of the f5 gröbner basis algorithm. *Journal of Symbolic Computation*, 70:49 – 70, 2015. <https://doi.org/10.1016/j.jsc.2014.09.025>.
 - [BFSY05] Magali Bardet, Jean-Charles Faugère, Bruno Salvy, and Bo-Yin 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*, pages 1–17, 2005. <https://hal.archives-ouvertes.fr/hal-01486845>.
 - [BG06] Olivier Billet and Henri Gilbert. Cryptanalysis of rainbow. In *Security and Cryptography for Networks – SCN 2006*, volume 4116 of *LNCS*, pages 336–347. Springer, 2006. https://doi.org/10.1007/11832072_23.
 - [BH98] Winfried Bruns and H. Jürgen Herzog. *Cohen-Macaulay Rings*, volume 39 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1998. <https://doi.org/10.1017/CB09780511608681>.
 - [Buc65] Bruno Buchberger. *Ein Algorithmus zum Auffinden der Basisselemente des Restklassenringes nach einem nulldimensionalen Polynomideal*. PhD thesis, Universität Innsbruck, 1965. https://www3.risc.jku.at/research/theorema/Groebner-Bases-Bibliography/gbbib_files/publication_706.pdf.
 - [CBD⁺09] Crystal Clough, John Baena, Jintai Ding, Bo-Yin Yang, and Ming-Shing Chen. Square, a new multivariate encryption scheme. In *Topics in Cryptology – CT-RSA 2009*, volume 5473 of *LNCS*, pages 252–264. Springer, 2009. https://doi.org/10.1007/978-3-642-00862-7_17.
 - [cCNY12] Chen-Mou cheng, Tung Chou, Ruben Niederhagen, and Bo-Yin Yang. Solving quadratic equations with xl on parallel architectures. In *Cryptographic Hardware and Embedded Systems – CHES 2012*, volume 7428 of *LNCS*, pages 356–373. Springer, 2012. https://doi.org/10.1007/978-3-642-33027-8_21.

- [CD05] Eduardo Cattani and Alicia Dickenstein. *Introduction to residues and resultants*, volume 14 of *Algorithms and Computation in Mathematics*. Springer, 2005. https://doi.org/10.1007/3-540-27357-3_1.
- [CFMR⁺17] A. Casanova, J.-C. Faugère, G. Macario-Rat, J. Patarin, L. Perret, and J. Ryckeghem. GeMSS: A great multivariate short signature. NIST PQC Submission, Université Pierre et Marie Curie, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>.
- [CFMR⁺19] A. Casanova, J.-C. Faugère, G. Macario-Rat, J. Patarin, L. Perret, and J. Ryckeghem. GeMSS: A great multivariate short signature (round 2 version). NIST PQC Submission for round 2, Université Pierre et Marie Curie, 2019. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions>.
- [CFMR⁺20] A. Casanova, J.-C. Faugère, G. Macario-Rat, J. Patarin, L. Perret, and J. Ryckeghem. GeMSS: A great multivariate short signature (round 2 version). NIST PQC Submission for round 3, Université Pierre et Marie Curie, 2020. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>.
- [CG19] Alessio Caminata and Elisa Gorla. The complexity of minrank. *Computing Research Repository*, Symbolic Computation, Cryptography and Security(1905.02682), 2019. <http://arxiv.org/abs/1905.02682>.
- [CGMT02] Nicolas Courtois, Louis Goubin, Willi Meier, and Jean-Daniel Tacier. Solving underdefined systems of multivariate quadratic equations. In *Public Key Cryptography – PKC 2002*, volume 2274 of *LNCS*, pages 211–227. Springer, 2002. https://doi.org/10.1007/3-540-45664-3_15.
- [CHR⁺16] Ming-Shing Chen, Andreas Hülsing, Joost Rijneveld, Simona Samardjiska, and Peter Schwabe. From 5-pass MQ-based identification to MQ-based signatures. In *Advances in Cryptology – ASIACRYPT 2016*, volume 10032 of *LNCS*, pages 135–165. Springer, 2016. https://doi.org/10.1007/978-3-662-53890-6_5.
- [CHR⁺17] Ming-Shing Chen, Andreas Hülsing, Joost Rijneveld, Simona Samardjiska, and Peter Schwabe. MQDSS specifications. NIST PQC Submission, Radboud University, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-3-submissions>.

- [quantum-cryptography/round-1-submissions](https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions).
- [CHR⁺19] Ming-Shing Chen, Andreas Hulsing, Joost Rijneveld, Simona Samardjiska, and Peter Schwabe. MQDSS specifications. NIST PQC Submission for round 2, Radboud University, 2019. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions>.
- [CJL⁺16] 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. <https://www.nist.gov/publications/report-post-quantum-cryptography>.
- [CKPS00] 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. https://doi.org/10.1007/3-540-45539-6_27.
- [CLO15] David A. Cox, John Little, and Donal O’Shea. *Ideals, Varieties, and Algorithms*. Graduate Texts in Mathematics. Springer, 2015. <https://doi.org/10.1007/978-3-319-16721-3>.
- [CNL⁺20] Jiahui Chen, Jianting Ning, Jie Ling, Terry Shue Chien Lau, and Yacheng Wang. A new encryption scheme for multivariate quadratic systems. *Theoretical Computer Science*, 809:372–383, 2020. <https://doi.org/10.1016/j.tcs.2019.12.032>.
- [Cop94] Don Coppersmith. Solving homogeneous linear equations over $\text{gf}(2)$ via block wiedemann algorithm. *Mathematics of Computation*, 62(205):333–350, 1994. <https://doi.org/10.2307/2153413>.
- [CP12] Jean-Jacques Quisquater Christophe Petit. On polynomial systems arising from a weil descent. In *Advances in Cryptology – ASIACRYPT 2012*, volume 7658 of *LNCS*, pages 451–466. Springer, 2012. https://doi.org/10.1007/978-3-642-34961-4_28.
- [CST18] Ryann Cartor and Daniel Smith-Tone. EFLASH: A new multivariate encryption scheme. In *Selected Areas in Cryptography – SAC 2018*, volume 11349 of *LNCS*, pages 281–299. Springer, 2018. https://doi.org/10.1007/978-3-030-10970-7_13.

- [CSTV17] Daniel Cabarcas, Daniel Smith-Tone, and Javier A. Vervel. Key recovery attack for ZHFE. In *Post-Quantum Cryptography 2017*, volume 10346 of *LNCS*, pages 289–308. Springer, 2017. https://doi.org/10.1007/978-3-319-59879-6_17.
- [CSV93a] Don Coppersmith, Jacques Stern, and Serge Vaudenay. Attacks on the birational permutation signature schemes. In *Advances in Cryptology - CRYPTO '93*, volume 773 of *LNCS*, pages 435–443. Springer, 1993. https://doi.org/10.1007/3-540-48329-2_37.
- [CSV93b] Don Coppersmith, Jacques Stern, and Serge Vaudenay. Attacks on the birational permutation signature schemes. In *Advances in Cryptology - CRYPTO'93*, volume 773 of *LNCS*, pages 435–443. Springer, 1993. https://doi.org/10.1007/3-540-48329-2_37.
- [CW90] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. *Journal of Symbolic Computation*, 9(3):251–280, 1990. [https://doi.org/10.1016/S0747-7171\(08\)80013-2](https://doi.org/10.1016/S0747-7171(08)80013-2).
- [DCP⁺17a] Jintai Ding, Ming-Shing Chen, Albrecht Petzoldt, Dieter Schmidt, and Bo-Yin Yang. Gui. NIST PQC Submission, University of Cincinnati, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>.
- [DCP⁺17b] Jintai Ding, Ming-Shing Chen, Albrecht Petzoldt, Dieter Schmidt, and Bo-Yin Yang. Rainbow - Algorithm Specification and Documentation. NIST PQC Submission, University of Cincinnati, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>.
- [DCP⁺19] Jintai Ding, Ming-Shing Chen, Albrecht Petzoldt, Dieter Schmidt, and Bo-Yin Yang. Rainbow - Algorithm Specification and Documentation (The 2nd Round Proposal). NIST PQC Submission for round 2, University of Cincinnati, 2019. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions>.
- [DCP⁺20] Jintai Ding, Ming-Shing Chen, Albrecht Petzoldt, Dieter Schmidt, and Bo-Yin Yang. Modified Parameters of Rainbow in Response to a Refind Analysis of the Rainbow Band Separation Attack by the NIST Team and the Recent New MinRank Attacks. NIST PQC Submission for round 3, University of Cincinnati, 2020. <https://csrc.nist.gov/projects/post->

- [quantum-cryptography/round-3-submissions](#).
- [DDY⁺08] Jintai Ding, Vivien Dubois, Bo-Yin Yang, Owen Chia-Hsin Chen, and Chen-Mou Cheng. Could SFLASH be repaired? In *International Colloquium on Automata, Languages and Programming – ICALP 2008*, volume 5126 of *LNCS*, pages 691–701. Springer, 2008. https://doi.org/10.1007/978-3-540-70583-3_56.
- [DFSS07] Vivien Dubois, Pierre-Alain Fouque, Adi Shamir, and Jacques Stern. Practical cryptanalysis of sflash. In *Advances in Cryptology – CRYPTO 2007*, volume 4622 of *LNCS*, pages 1–12. Springer, 2007. https://doi.org/10.1007/978-3-540-74143-5_1.
- [DG10] Vivien Dubois and Nicolas Gama. The degree of regularity of hfe systems. In *Advances in Cryptology – ASIACRYPT 2010*, volume 6477 of *LNCS*, pages 557–576. Springer, 2010. https://doi.org/10.1007/978-3-642-17373-8_32.
- [DH76] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, 1976. <https://doi.org/10.1109/TIT.1976.1055638>.
- [DH11] Jintai Ding and Timothy J. Hodges. Inverting HFE system is quasipolynomial for all fields. In *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *LNCS*, pages 724–742. Springer, 2011. https://doi.org/10.1007/978-3-642-22792-9_41.
- [DK12] Jintai Ding and Thorsten Kleinjung. Degree of regularity for hfe-. *Journal of Math-for-Industry*, 4(2012B-3):97–104, 2012. <http://j-mi.org/articles/view/272>.
- [DPPST18] Jintai Ding, Ray Perlner, Albrecht Petzoldt, and Daniel Smith-Tone. Improved cryptanalysis of hfev- via projection. In *Post-Quantum Cryptography – PQCrypto2018*, volume 10786 of *LNCS*, pages 375–395. Springer, 2018. https://doi.org/10.1007/978-3-319-79063-3_18.
- [DS05] 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. https://doi.org/10.1007/11496137_12.
- [DY13] Jintai Ding and Bo-Yin Yang. Degree of regularity for HFEv and HFEv-.

- In *Post-Quantum Cryptography 2013*, volume 7932 of *LNCS*, pages 52–66. Springer, 2013. https://doi.org/10.1007/978-3-642-38616-9_4.
- [DYC⁺08] Jintai Ding, Bo-Yin Yang, Chia-Hsin Owen Chen, Ming-Shing Chen, and Chen-Mou Cheng. New differential-algebraic attacks and reparametrization of rainbow. In *Applied Cryptography and Network Security – ACNS 2008*, volume 5037 of *LNCS*, pages 242–257. Springer, 2008. https://doi.org/10.1007/978-3-540-68914-0_15.
- [Eis95] David Eisenbud. *Commutative Algebra with a View Toward Algebraic Geometry*, volume 150 of *Graduate Texts in Mathematics*. Springer-Verlag, 1995. <https://doi.org/10.1007/978-1-4612-5350-1>.
- [Fau99] Jean-Charles Faugère. A new efficient algorithm for computing Gröbner bases (F4). *Journal of Pure and Applied Algebra*, 139(1):61 – 88, 1999. [https://doi.org/10.1016/S0022-4049\(99\)00005-5](https://doi.org/10.1016/S0022-4049(99)00005-5).
- [Fau02] Jean Charles Faugère. A new efficient algorithm for computing Gröbner Bases without reduction to zero (F5). In *Proceedings of the 2002 international symposium on Symbolic and algebraic computation*, pages 75–83. ACM, 2002. <https://doi.org/10.1145/780506.780516>.
- [FDS10] 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. <http://doi.acm.org/10.1145/1837934.1837984>.
- [FDS11] 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. <https://doi.org/10.1016/j.jsc.2010.10.014>.
- [FDS13] 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. <https://doi.org/10.1016/j.jsc.2013.03.004>.
- [FGS05] Pierre-Alain Fouque, Louis Granboulan, and Jacques Stern. Differential cryptanalysis for multivariate schemes. In *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *LNCS*, pages 341–353. Springer, 2005.

- https://doi.org/10.1007/11426639_20.
- [FH02] Ralf Fröberg and Joachim Hollman. Hilbert series for ideals generated by generic forms. *Journal of Symbolic Computation*, 17(2):149–157, 2002. <https://doi.org/10.1006/jsco.1994.1008>.
- [FJ03] Jean-Charles Faugère and Antoine Joux. Algebraic cryptanalysis of hidden field equation (hfe) cryptosystems using gröbner bases. In *Advances in Cryptology – CRYPTO 2003*, volume 2729 of *LNCS*, pages 44–60. Springer, 2003. https://doi.org/10.1007/978-3-540-45146-4_3.
- [FKI⁺20] Hiroki Furue, Koha Kinjo, Yasuhiko Ikematsu, Yacheng Wang, and Tsuyoshi Takagi. A structural attack on block-anti-circulant uov at SAC 2019. In *Post-Quantum Cryptography*, volume 12100 of *LNCS*, pages 323–339. Springer, 2020. https://doi.org/10.1007/978-3-030-44223-1_18.
- [FLdVP08] Jean-Charles Faugère, Françoise Levy-dit Vehel, and Ludovic Perret. Cryptanalysis of minrank. In *Advances in Cryptology – CRYPTO 2008*, volume 5157 of *LNCS*, pages 280–296. Springer, 2008. https://doi.org/10.1007/978-3-540-85174-5_16.
- [FPPR11] Jean-Charles Faugère, Ludovic Perret, Christophe Petit, and Guénaél Renault. New subexponential algorithms for factoring in $sl(2, \mathbb{F}_q)$. *Cryptology ePrint Archive*, Report 2011/598, 2011. <https://eprint.iacr.org/2011/598>.
- [FPPR12] Jean-Charles Faugère, Ludovic Perret, Christophe Petit, and Guénaél Renault. Improving the complexity of index calculus algorithms in elliptic curves over binary fields. In *Advances in Cryptology – EURO-CRYPT 2012*, volume 7237 of *LNCS*, pages 27–44. Springer, 2012. https://doi.org/10.1007/978-3-642-29011-4_4.
- [FPR17] J.-C. Faugère, L. Perret, and J. Ryckeghem. DualModeMS: A dual mode for multivariate-based signature. NIST PQC Submission, Université Pierre et Marie Curie, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>.
- [FR94] G. Frey and H.-G. Ruck. A remark concerning m -divisibility and the discrete logarithm in the divisor class group of curves. *Math. Comp.*, 62:865–874, 1994. <https://doi.org/10.2307/2153546>.
- [Frö85] Ralf Fröberg. An inequality for Hilbert series of graded algebras. *Math-*

- ematica Scandinavica*, 56:117–144, 1985. <https://doi.org/10.7146/math.scand.a-12092>.
- [FSEDS11] 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, April 2011. <https://doi.org/10.1016/j.jsc.2010.10.014>.
- [GC00a] 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. https://doi.org/10.1007/3-540-44448-3_4.
- [GC00b] 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. https://doi.org/10.1007/3-540-44448-3_4.
- [GM05] Patrizia Gianni and Teo Mora. Algebraic solution of systems of polynomial equations using gröbner bases. In *Applied Algebra, Algebraic Algorithms and Error-Correcting Codes — AAEECC1987*, volume 356 of *LNCS*, pages 247–257. Springer, 2005. https://doi.org/10.1007/3-540-51082-6_83.
- [Gro96] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing — STOC '96*, STOC, pages 212–219. ACM, 1996. <https://doi.org/10.1145/237814.237866>.
- [HKY15] Ming-Deh A. Huang, Michiel Kisters, and Sze Ling Yeo. Last fall degree, hfe and weil descent attacks on ecdlp. In *Advances in Cryptology — CRYPTO 2015*, volume 9215 of *LNCS*, pages 58–600. Springer, 2015. https://doi.org/10.1007/978-3-662-47989-6_28.
- [HKYY18] Ming-Deh A. Huang, Michiel Kisters, Yun Yang, and Sze Ling Yeo. On the last fall degree of zero-dimensional weil descent systems. *Journal of Symbolic Computation*, 87:207–226, 2018. <https://doi.org/10.1016/j.jsc.2017.08.002>.
- [IPST⁺18] Yasuhiko Ikematsu, Ray Perlner, Daniel Smith-Tone, Tsuyoshi Takagi, and Jeremy Vates. HFERP - a new multivariate encryption scheme. In *Post-Quantum Cryptography — PQCrypto 2018*, volume 10786 of *LNCS*, pages

- 396–416. Springer, 2018. https://doi.org/10.1007/978-3-319-79063-3_19.
- [JHPS14] Timothy J. Hodges, Christophe Petit, and Jacob Schlather. First fall degree and weil descent. *Finite Fields and Their Applications*, 30:155–177, 2014. <https://doi.org/10.1016/j.ffa.2014.07.001>.
- [JV18] Antoine Joux and Vanessa Vitse. A crossbred algorithm for solving boolean polynomial systems. In *Number-Theoretic Methods in Cryptology – NuTMiC 2017*, volume 10737 of *LNCS*, pages 3–21. Springer, 2018. https://doi.org/10.1007/978-3-319-76620-1_1.
- [Kob87] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of Computation*, 48:203–209, 1987. <https://doi.org/10.1090/S0025-5718-1987-0866109-5>.
- [KPG99] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar signature schemes. In *Advances in Cryptology – EUROCRYPT ’99*, volume 1592 of *LNCS*, pages 206–222. Springer, 1999. https://doi.org/10.1007/3-540-48910-X_15.
- [KPG03] Aviad Kipnis, Jacques Patarin, and Louis Goubin. Unbalanced oil and vinegar schemes – extended version. In *Advances in Cryptology – EUROCRYPT’99*, volume 1592 of *LNCS*, pages 1–18. Springer, 2003. <http://www.goubin.fr/papers/OILLONG.PDF>.
- [KS98] Aviad Kipnis and Adi Shamir. Cryptanalysis of the oil and vinegar signature scheme. In *Advances in Cryptology – CRYPTO’98*, volume 1462 of *LNCS*, pages 257–266. Springer, 1998. <https://doi.org/10.1007/BFb0055733>.
- [KS99] 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. https://doi.org/10.1007/3-540-48405-1_2.
- [KZ20] Daniel Kales and Greg Zaverucha. An attack on some signature schemes constructed from five-pass identification schemes. *Cryptology ePrint Archive*, 2020(837), 2020. <https://eprint.iacr.org/2020/837>.
- [Lan02] Serge Lang. *Algebra - Revised Third Edition*. Graduate Texts in Mathematics. Springer, 2002. <https://doi.org/10.1007/978-1-4613-0041-0>.

- [Laz83] D. Lazard. Gröbner bases, Gaussian elimination and resolution of systems of algebraic equations. In *European Conference on Computer Algebra - EUROCAL 1983*, volume 162 of *LNCS*, pages 146–156. Springer, 1983. https://doi.org/10.1007/3-540-12868-9_99.
- [Lev44] Kenneth Levenberg. A method for the solution of certain non-linear problems in least squares. *Quarterly of Applied Mathematics*, 2(2):164–168, 1944. <https://www.jstor.org/stable/43633451>.
- [Mar63] Donald W. Marquardt. An algorithm for least-squares estimation of non-linear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963. <https://doi.org/10.1137/0111030>.
- [MHT13] Hiroyuki Miura, Yasufumi Hashimoto, and Tsuyoshi Takagi. Extended algorithm for solving underdefined multivariate quadratic equations. In *Post-Quantum Cryptography – PQCrypto 2013*, volume 7932 of *LNCS*, pages 118–135. Springer, 2013. https://doi.org/10.1007/978-3-642-38616-9_8.
- [MI88] Tsutomu Matsumoto and Hideki Imai. Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In *Advances in Cryptology – EUROCRYPT ’88*, volume 330 of *LNCS*, pages 419–453. Springer, 1988. https://doi.org/10.1007/3-540-45961-8_39.
- [Moh11] Wael Said Abdelmageed Mohamed. *Improvements for the XL algorithm with Applications to Algebraic Cryptanalysis*. PhD thesis, Technical University of Darmstadt, 2011. https://tuprints.ulb.tu-darmstadt.de/2621/4/WST_Diss.pdf.
- [Mor78] Jorge J. Moré. The levenberg-marquardt algorithm: Implementation and theory. In *Numerical Analysis*, volume 630 of *LNM*, pages 105–116. Springer, 1978. <https://doi.org/10.1007/BFb0067700>.
- [MPST14] Dustin Moody, Ray Perlner, and Daniel Smith-Tone. An asymptotically optimal structural attack on the ABC multivariate encryption scheme. In *Post-Quantum Cryptography – PQCrypto 2014*, volume 8772 of *LNCS*, pages 180–196. Springer, 2014. https://doi.org/10.1007/978-3-319-11659-4_11.
- [MS05] Ezra Miller and Bernd Sturmfels. *Combinatorial Commutative Algebra*, volume 227 of *Graduate Texts in Mathematics*. Springer, 2005. <https://doi.org/10.1007/978-1-4939-9987-1>.

- [//doi.org/10.1007/b138602](https://doi.org/10.1007/b138602).
- [NIW⁺20] Shuhei Nakamura, Yasuhiko Ikematsu, Yacheng Wang, Jintai Ding, and Tsuyoshi Takagi. New complexity estimation on the rainbow-band-separation attack. *Cryptology ePrint Archive*, 2020(703), 2020. <https://eprint.iacr.org/2020/703>.
- [Pat95] Jacques Patarin. Cryptanalysis of the Matsumoto and Imai public key scheme of Eurocrypt'88. In *Advances in Cryptology – CRYPTO'95*, volume 963 of *LNCS*, pages 248–261. Springer, 1995. https://doi.org/10.1007/3-540-44750-4_20.
- [Pat96] 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. https://doi.org/10.1007/3-540-68339-9_4.
- [PBD14] Jaiberth Porras, John Baena, and Jintai Ding. ZHFE, a new multivariate public key encryption scheme. In *Post-Quantum Cryptography – PQCrypto 2014*, volume 8772 of *LNCS*, pages 229–245. Springer, 2014. https://doi.org/10.1007/978-3-319-11659-4_14.
- [PCDY17] Albrecht Petzoldt, Ming-Shing Chen, Jintai Ding, and Bo-Yin Yang. HMEv - an efficient multivariate signature scheme. In *PQCrypto 2017 : Post-Quantum Cryptography*, volume 10346 of *LNCS*, pages 205–223. Springer, 2017. https://doi.org/10.1007/978-3-319-59879-6_12.
- [PCG01a] Jacques Patarin, Nicolas Courtois, and Louis Goubin. FLASH, a fast multivariate signature algorithm. In *Topics in Cryptology – CT-RSA 2001*, volume 2020 of *LNCS*, pages 298–307. Springer, 2001. https://doi.org/10.1007/3-540-45353-9_22.
- [PCG01b] 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. https://doi.org/10.1007/3-540-45353-9_21.
- [PCY⁺15] Albrecht Petzoldt, Ming-Shing Chen, Bo-Yin Yang, Chengdong Tao, and Jintai Ding. Design principles for HFEv- based multivariate signature schemes. In *Advances in Cryptology – ASIACRYPT 2015*, volume 9452 of *LNCS*, pages 311–334. Springer, 2015. <https://doi.org/10.1007/978->

- 3-662-48797-6_14.
- [Pet17] Albrecht Petzoldt. On the complexity of the hybrid approach on HFEv-. *Cryptology ePrint Archive*, 2017(1135), 2017. <https://eprint.iacr.org/2017/1135>.
- [PPST18] 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. https://doi.org/10.1007/978-3-319-72565-9_18.
- [PST20] Ray Perlner and Daniel Smith-Tone. Rainbow band separation is better than we thought. *Cryptology ePrint Archive*, 2020(702), 2020. <https://eprint.iacr.org/2020/702>.
- [RGSJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, 1979. <https://dl.acm.org/doi/book/10.5555/574848>.
- [RSA78] 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. <https://doi.org/10.1145/359340.359342>.
- [SDP16] Alan Szepieniec, Jintai Ding, and Bart Preneel. Extension Field Cancellation: A new central trapdoor for multivariate quadratic systems. In *Post-Quantum Cryptography – PQCrypto 2016*, volume 9606 of LNCS, pages 182–196. Springer, 2016. https://doi.org/10.1007/978-3-319-29360-8_12.
- [Sho97] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal on Computing*, 26(5):1484–1509, 1997. <https://doi.org/10.1137/S0036144598347011>.
- [SP17] Alan Szepieniec and Bart Preneel. Short solutions to nonlinear systems of equations. In *Number-Theoretic Methods in Cryptology*, volume 10737 of LNCS, pages 71–90. Springer, 2017. https://doi.org/10.1007/978-3-319-76620-1_5.
- [SPK17] Kyung-Ah Shim, Cheol-Min Park, and Namhun Koo. An existential unforgeable signature scheme based on multivariate quadratic equations. In *Advances in Cryptology - ASIACRYPT 2017*, volume 10624 of LNCS, pages 37–64. Springer, 2017. https://doi.org/10.1007/978-3-319-70694-8_2.

- [SPKK17] Kyung-Ah Shim, Cheol-Min Park, Aeyoung Kim, and Namhun Koo. HiMQ-3: A high speed signature scheme based on multivariate quadratic equations. NIST PQC Submission, National Institute for Mathematical Sciences, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>.
- [SSH11] Koichi Sakumoto, Taizo Shirai, and Harunaga Hiwatari. Public-key identification schemes based on multivariate quadratic polynomials. In *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *LNCS*, pages 706–723. Springer, 2011. https://doi.org/10.1007/978-3-642-22792-9_40.
- [Sto00] Arne Storjohann. *Algorithms for Matrix Canonical Forms*. PhD thesis, Swiss Federal Institute of Technology, 2000. <https://cs.uwaterloo.ca/~astorjoh/diss2up.pdf>.
- [Sto10] Andrew James Stothers. *On the Complexity of Matrix Multiplication*. PhD thesis, University of Edinburgh, 2010. <https://www.maths.ed.ac.uk/sites/default/files/atoms/files/stothers.pdf>.
- [Str69] Volker Strassen. Gaussian elimination is not optimal. *Numerische Mathematik*, 13(4):354–356, 1969. <https://doi.org/10.1007/BF02165411>.
- [TCS15] Satoshi Tanaka, Chen-Mou Cheng, and Kouichi Sakurai. Evaluation of solving time for multivariate quadratic equation system using xl algorithm over small finite fields on gpu. In *Mathematics and Computing*, volume 139 of *Springer Proceedings in Mathematics & Statistics*, pages 349–361. Springer, 2015. https://doi.org/10.1007/978-81-322-2452-5_24.
- [TDTD13] Chengdong Tao, Adama Diene, Shaohua Tang, and Jintai Ding. Simple matrix scheme for encryption. In *PQCrypto 2013: Post-Quantum Cryptography 2013*, volume 7932 of *LNCS*, pages 231–242. Springer, 2013. https://doi.org/10.1007/978-3-642-38616-9_16.
- [Tho02] Emmanuel Thomé. Subquadratic computation of vector generating polynomials and improvement of the block wiedemann algorithm. *Journal of Symbolic Computation*, 33(5):757 – 775, 2002. <https://doi.org/10.1006/jSCO.2002.0533>.
- [Tho13] Enrico Thomae. *About the Security of Multivariate Quadratic Public Key Schemes*. PhD thesis, Ruhr-Universität Bochum, 2013. <https://www.cits.ruhr-uni-bochum.de/personen/thomae.html>.

- [TW12] Enrico Thomae and Christopher Wolf. Solving underdetermined systems of multivariate quadratic equations revisited. In *Public Key Cryptography – PKC2012*, volume 7293 of *LNCS*, pages 156–171. Springer, 2012. https://doi.org/10.1007/978-3-642-30057-8_10.
- [VBC⁺19] Javier Verbel, John Baena, Daniel Cabarcas, Ray Perlner, and Daniel Smith-Tone. On the complexity of “superdetermined” minrank instances. In *Post Quantum Cryptography – PQCrypto 2019*, volume 11505 of *LNCS*, pages 167–186. Springer, 2019. https://doi.org/10.1007/978-3-030-25510-7_10.
- [VST17] Jeremy Vates and Daniel Smith-Tone. Key recovery attack for all parameters of HFE-. In *Post-Quantum Cryptography 2017*, volume 10346 of *LNCS*, pages 272–288. Springer, 2017. https://doi.org/10.1007/978-3-319-59879-6_16.
- [War20] Ward. Improved cryptanalysis of uov and rainbow. *Cryptology ePrint Archive*, 2020(1343):1–26, 2020. <https://eprint.iacr.org/2020/1343>.
- [WIDT18] Yacheng Wang, Yasuhiko Ikematsu, Dung Hoang Duong, and Tsuyoshi Takagi. Efficient decryption algorithms for extension field cancellation type encryption schemes. In *Australasian Conference on Information Security and Privacy – ACISP 2018*, *LNCS*, pages 487–501. Springer, 2018. https://doi.org/10.1007/978-3-319-93638-3_28.
- [WIDT19] Yacheng Wang, Yasuhiko Ikematsu, Dung Hoang Duong, and Tsuyoshi Takagi. The secure parameters and efficient decryption algorithm for multivariate public key cryptosystem etc. *IEICE Transactions on Fundamentals of Electronics, Communication and Computer Sciences*, E102-A(9):1028–1036, 2019. <https://doi.org/10.1587/transfun.E102.A.1028>.
- [Wie86] Douglas H. Wiedemann. Solving sparse linear equations over finite fields. *IEEE Transactions on Information Theory*, IT-32(1):54–62, 1986. <https://doi.org/10.1109/TIT.1986.1057137>.
- [Wil12] Virginia Vassilevska Williams. Multiplying matrices faster than coppersmith-winograd. In *Proceedings of the forty-fourth annual ACM symposium on Theory of computing – STOC ’12*, pages 887–898. ACM, 2012. <https://doi.org/10.1145/2213977.2214056>.
- [WINT20] Yacheng Wang, Yasuhiko Ikematsu, Shuhei Nakamura, and Tsuyoshi Tak-

- agi. Revisiting the minrank problem on multivariate cryptography. In *The 21st World Conference on Information Security Application – WISA2020*, LNCS. Springer, 2020. to appear.
- [Wol04] Christopher Wolf. Efficient cryptanalysis of RSE(2)PKC and RSSE(2)PKC. In *Security in Communication Networks – SCN 2004*, volume 3352 of *LNCS*, pages 294–309. Springer, 2004. https://doi.org/10.1007/978-3-540-30598-9_21.
- [WP05] Christopher Wolf and Bart Preneel. Equivalent keys in HFE, C*, and variations. In *Progress in Cryptology – Mycrypt 2005*, volume 3715 of *LNCS*, pages 33–49. Springer, 2005. https://doi.org/10.1007/11554868_4.
- [WPSV17] Ward, Bart Preneel, Alan Szepieniec, and Frederik Vercauteren. LUOV, signature scheme proposal for NIST PQC project. NIST PQC Submission, imec-COSIC KU Leuven, 2017. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submissions>.
- [WPSV19] Ward, Bart Preneel, Alan Szepieniec, and Frederik Vercauteren. LUOV, signature scheme proposal for NIST PQC project (round 2 version). NIST PQC Submission for round 2, imec-COSIC KU Leuven, 2019. <https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions>.
- [Yas18] Takanori Yasuda. Multivariate encryption schemes based on the constrained MQ problem. In *Provable Security – ProvSec 2018*, volume 11192 of *LNCS*, pages 129–146. Springer, 2018. https://doi.org/10.1007/978-3-030-01446-9_8.
- [YC04] Bo-Yin Yang and Jiun-Ming Chen. Theoretical analysis of XL over small fields. In *Information Security and Privacy – ACISP 2004*, volume 3108 of *LNCS*, pages 277–288. Springer, 2004. https://doi.org/10.1007/978-3-540-27800-9_24.
- [YDH⁺15a] Takanori Yasuda, Xavier Dahan, Yun-Ju Huang, Tsuyoshi Takagi, and Kouichi Sakurai. Mq challenge: Hardness evaluate of solving multivariate quadratic problems. *Cryptology ePrint Archive*, 2015(275), 2015. <https://eprint.iacr.org/2015/275.pdf>.
- [YDH⁺15b] Takanori Yasuda, Xavier Dahan, Yun-Ju Huang, Tsuyoshi Takagi, and Kouichi Sakurai. MQ challenge: Hardness evaluation of solving multi-

- variate quadratic problems. *Cryptology ePrint Archive*, 2015(275), 2015. <https://eprint.iacr.org/2015/275>.
- [YS16] Takanori Yasuda and Kouichi Sakurai. A multivariate encryption scheme with Rainbow. In *Information and Communications Security – ICICS 2015*, volume 9543 of *LNCS*, pages 236–251. Springer, 2016. https://doi.org/10.1007/978-3-319-29814-6_19.
- [YWT20] Takanori Yasuda, Yacheng Wang, and Tsuyoshi Takagi. Multivariate encryption schemes based on polynomial equations over real numbers. In *Post-Quantum Cryptography – PQCrypto 2020*, volume 12100 of *LNCS*, pages 402–421. Springer, 2020. https://doi.org/10.1007/978-3-030-44223-1_22.