University of Louisville

ThinkIR: The University of Louisville's Institutional Repository

Electronic Theses and Dissertations

8-2018

Developments in multivariate post quantum cryptography.

Jeremy Robert Vates
University of Louisville

DEVELOPMENTS IN MULTIVARIATE POST QUANTUM CRYPTOGRAPHY

By

Jeremy Robert Vates
B.S., Marian University, 2013
M.A., University of Louisville, 2015

A Dissertation
Submitted to the Faculty of the
College of Arts and Sciences of the University of Louisville
in Partial Fulfillment of the Requirements
for the Degree of

Doctor of Philosophy
in
Applied and Industrial Mathematics

Department of Mathematics
University of Louisville
Louisville, KY

August 2018

DEVELOPMENTS IN MULTIVARIATE POST QUANTUM CRYPTOGRAPHY

Submitted by

Jeremy Robert Vates

A Dissertation Approved on

April 23, 2018

by the Following Dissertation Committee:

Dr. Daniel Smith-Tone, Dissertation Director

Dr. David Brown

Dr. Hamid Kulosman

Dr. Jinjia Li

Dr. Steven Seif

DEDICATION

To my son, Gideon, who motivated me.

To my wife, Danielle, who supported me.

To my parents, Joseph and Kelly, who always believed in me.

ACKNOWLEDGEMENTS

I would like to begin by thanking my advisor and dissertation director, Dr. Daniel Smith-Tone for his unending support throughout my graduate career. His insight and encouragement formed the foundation for my love of post quantum cryptography. He always pushed me to improve myself in professional development and to pursue my passions in mathematics and computer science. Without him, I would not be the mathematician I am today.

I would like to thank the members of my dissertation committee, Dr. David Brown, Dr. Hamid Kulosman, Dr. Jinjia Li, and Dr. Steven Seif. I thank each of them for taking the time to read into my discipline in order to provide appreciated criticism of this dissertation and to help me make it the best it can be. I am grateful for their continuous support.

I would like to thank the faculty of the College of Arts and Science at Marian University. Specificially, I would like to extend a thanks to Dr. Ben Allgeier, Dr. Carl Lecher, and Dr. Rod Macrae. It is thanks to Dr. Lecher and Dr. Macrae for encouraging me to do undergraduate research and learn where my passion of mathematics and computer programming can play a role in the advancement and sustainability of mankind. It is thanks to Dr. Allgeier for introducing me to the beauty of higher mathematics and encouraging me to pursue a doctorate. Without him as a mentor and teacher, I would not see the beauty and creativity that is mathematics. I would also like to extend a thanks to Marian University as a whole. This university invested their time and money into me as a student and still stands as my standard for how a university should invest in their students. Without Marian, I would not be the student I am today, and for that, I am eternally grateful.

I would also like to thank the many people that contributed in numerous different ways that led me to where I am today. Dr. Jintai Ding for his expertise in Post-Quantum Crpytography and assistance in my professional development. PQCrypto, an annual conference where brilliant individuals gather to share and further the science. It is there where my passion for research was discovered. Fellow graduate students at the University of Louisville, whose companionship and collaboration brightened my graduate school experience.

Finally, I would like to thank God for where I am today, for my passion of mathematics, and for my ability to see the beauty that is math. It is my desire to continue to pursue, as Galileo said, ”Mathematics... the alphabet in which God has written the universe.”

ABSTRACT

DEVELOPMENTS IN MULTIVARIATE POST QUANTUM CRYPTOGRAPHY

Jeremy Robert Vates

April 23, 2018

Ever since Shor's algorithm was introduced in 1994, cryptographers have been working to develop cryptosystems that can resist known quantum computer attacks. This push for quantum attack resistant schemes is known as post quantum cryptography. Specifically, my contributions to post quantum cryptography has been to the family of schemes known as Multivariate Public Key Cryptography (MPKC), which is a very attractive candidate for digital signature standardization in the post quantum collective for a wide variety of applications. In this document I will be providing all necessary background to fully understand MPKC and post quantum cryptography as a whole. Then, I will walk through the contributions I provided in my publications relating to differential security proofs for HFEv and HFEv$^-$, key recovery attack for all parameters of HFEm, and my newly proposed multivariate encryption scheme, HFERP.

TABLE OF CONTENTS

CHAPTER 1
INTRODUCTION

Since humans discovered the ability to write, there has been a need for protecting information from unwanted eyes. For centuries, methods have been devised to secure written information. These methods could manipulate the characters the information was written in, like a simple substitution cypher, or they could manipulate the materials the information was written on, like the scytale cypher. However, as technology advanced, so did the complexity of feasible cryptosystems.

Most of history's use of cryptography has depended on the use of a shared secret. In order for two parties to communicate securely, they would need to have a predetermined meeting in which to exchange a shared secret that would be the basis for their secure communication. There are multiple cryptosystems developed based upon this topic like the Hill Cipher, see [29], and the infamous Enigma, see [43].

As the need to communicate securely over vast distances became a reality, key distribution became an issue. This was addressed in the 1970's by the Diffie-Hellman key exchange, see [15]. They were able to introduce a technique that depended on exponentiation that allowed the generation of a shared secret remotely in a secure manner. This technique was one of the first indications that it was possible to secure information without the need of a shared secret. Most notably, James Ellis was the pioneer of this field of study in 1975, which became known as public key cryptography, see [60].

Public key cryptography quickly drew the attention of cryptography researchers. New schemes were developed to take advantage of the ideas introduced by Diffie, Hellman, and Ellis, most notably RSA created by Rivest, Shamir, and Adleman, see [58]. This system was one of the first schemes to fully utilize the concept of public key cryptography and is still one of the most relied upon in modern technology. One can see the inspiration of RSA from the Diffie-Hellman Key Exchange as both utilize the one way function of exponentiation. However, the security of RSA is based on the difficulty of factoring large numbers rather than the discrete log problem which is the basis of security of the Diffie-Hellman Key Exchange.

Recently, the focus has shifted among many to another new area of cryptography. To understand this shift, one must understand the importance of quantum computing and Peter Shor. Advancements in quantum computing have shown that quantum computing devices are no longer the pyrite of the sci-fi genre. It is not known when such devices will be a realistic threat to cryptography, but many companies, organizations, and governments are currently in a rush to develop such technology, see [31] and [65] . Regardless of when, the inevitability is sufficient motivation for post-quantum security standards as seen by NIST's call for standardization, see [44].

A common question is posed by those new to the field: If one does not have a quantum computer to work with, how does one develop quantum resistant cryptography? The answer begins with Peter Shor. In the 1990's, Shor created an algorithm to factor and compute discrete logarithms in polynomial time on quantum computing device [62]. This is a problem for widely used schemes like R.S.A., whose security depends on the difficulty of factoring very large numbers. With the introduction of Shor's algorithms and the use of a quantum computing device, an individual can break RSA in polynomial time. An example of such a development is the most recent record of factorization using a quantum computer of 56153, see

[14]. Even though the scenario where quantum computers break through modern implementations of RSA is years away, the possibility of this occurring is unacceptable. Thus, post-quantum cryptography focuses on developing schemes where their security does not depend on problems that we know quantum computing devices can solve quickly.

One example of a family of such problems is based on solving systems of multivariate equations which forms the basis of Multivariate Public Key Cryptography (MPKC). This collection of schemes generate a system of quadratic equations to be used as the public key. Within this collection of schemes lives a subset of systems called "Big Field Schemes". Here, equations are generated by constructing a finite extension of a finite field and using a core map that lives in said extension. Then, by using Frobenius automorphisms and natural vector space isomorphisms, one can easily generate a system of quadratic equations in a method that allows efficient inversion. An outside observer would have to be able to solve a system of quadratic equations over a finite field, which is known to be NP-complete, see [27].

In this dissertation, I walk through all preliminary mathematics necessary to understand the construction and cryptanalysis of Multivariate Public Key Cryptography along with a basic introduction to public key cryptography as a whole. Then, I go into detail on three cryptosystems and their modifiers within MPKC: HFE, HFEv, and HFERP. I go into detail on the construction of each as well as a detailed walk through my contributions to the Post-Quantum Cryptography Community from my following publications:

- Key Recovery Attack for All Parameters of HFEm, see [66]
- On the Differential Security of the HFEv- Signature Primitive, see [11]
- HFERP - A New Multivarite Encryption Scheme, see [30]

CHAPTER 2
PRELIMINARIES

Within Multivariate Public Key Cryptography, my work has focused on a family of schemes called *big field* schemes. These schemes have a structure based within concepts taught in an Abstract Algebra course. For reference, I am including basic definitions, proofs, and toy examples of these structures.

2.1 Algebraic Preliminaries

To begin, we start with the definition of a ring and build up to a field.

Definition 2.1 (see [21]). *A **ring** R is a set together with two binary operators $+$ and $\times$ (called addition and multiplication) satisfying the following axioms (Note: we write $a \times b$ as ab for any $a, b \in R$ for convenience)*

1. *$(R, +)$ is an **abelian** group (group is commutative under the specified operation),*

2. *$\times$ is **associative**: $(a \times b) \times c = a \times (b \times c)\ \forall\ a, b, c\ \in R$,*

3. *the **distributive** law holds in R: $a(b + c) = ab + ac\ \forall\ a, b, c\ \in R$.*

It is worth noting that, throughout this text, we will be dealing with rings that have a multiplicative identity, a.k.a. $\exists\ a\ \in\ R$ such that $a \times b = b \times a = b\ \forall\ b \in\ R$. Normally, we denote the multiplicative identity as 1_R or 1 if the ring context is clear.

Before we move onto fields, it is worth defining the basics of homomorphisms and some ring theory.

Definition 2.2 (see [21])**.** *Let R and S be rings.*

- *A* ***ring homomorphism*** *is a map $\varphi : R \to S$ satisfying*

  $\varphi(a+b) = \varphi(a) + \varphi(b)$ *for all* $a, b \in R$.

  $\varphi(ab) = \varphi(a)\varphi(b)$ *for all* $a, b \in R$

- *The* ***kernel*** *of a ring homomorphism φ, denoted $ker(\varphi)$ is the set of elements of R that map to 0 in S.*

- *A bijective ring homomorphism is called an isomorphism. If there exists an isomorphism between rings R and S, we say they are* ***isomorphic*** *and this is denoted as $R \cong S$.*

After considering the construction of ring homomorphisms, the structure of its kernel is a logical leap on inquiry. These objects play a large role in ring theory, thus deserve their own name: Ideals.

Definition 2.3 (see [21])**.** *Let R be a ring, let I be a subset of R and let $r \in R$.*

1. $rI = \{ra \mid a \in I\}$ *and* $Ir = \{ar \mid a \in I\}$

2. *A subset I of R is a* ***left ideal*** *of R if*

   I is a subring of R, and

   I is closed under left multiplication by elements in R, i.e., $rI \subseteq I$ for all $r \in R$.

3. *A subset I of R is a* ***right ideal*** *of R if*

   I is a subring of R, and

   I is closed under right multiplication by elements in R, i.e., $Ir \subseteq I$ for all $r \in R$.

4. *A subset I that is both a left and right ideal is called an* ***ideal****, or a* ***two-sided ideal*** *for emphasis.*

THEOREM 2.1 (see [21]). ***(The First Isomorphism Theorem)*** *If $\varphi : R \to S$ is a homomorphism of rings, then the following are true:*

- *$ker(\varphi)$ is an ideal of R,*
- *the image of φ, $\varphi(R)$, is a subring of S, and*
- *$R/ker(\varphi) \cong \varphi(R)$*

Proof. Let $\varphi : R \to S$ be a homomorphism of rings R and S.

- To prove that $ker(\varphi)$ is an ideal of R, I will begin by showing it is non-empty then showing that the properties of an ideal, closed under addition and left and right multiplication, are held true in $ker(\varphi)$. Since φ is a ring homomorphism, we know that $\varphi(0_R) = 0_S$, thus showing that $ker(\varphi)$ is nonempty. Finally, let $k_1, k_2 \in ker(\varphi)$ and $r \in R$ and observe

$$\varphi(k_1 + k_2) = \varphi(k_1) = \varphi(k_2) = 0_S + 0_S = 0_S$$

$$\varphi(rk_1) = \varphi(r)\varphi(k_1) = \varphi(r) \cdot 0_S = 0_S$$

$$\varphi(k_1 r) = \varphi(k_1)\varphi(r) = 0_S \cdot \varphi(r) = 0_S$$

  Thus, $ker(\varphi)$ is an ideal of R.

- To show that $\varphi(R)$ is a subring of S, I will show that 1_S, the identity of S, is in $\varphi(R)$, and that $\varphi(R)$ is closed under subtraction and multiplication with respect to S. Let $s_1, s_2 \in \varphi(R)$. Thus, there exists $r_1, r_2 \in R$ such that $\varphi(r_1) = s_1$ and $\varphi(r_2) = s_2$. Observe

$$s_1 - s_2 = \varphi(r_1) - \varphi(r_2) = \varphi(r_1) + \varphi(-r_2) = \varphi(r_1 - r_2).$$

This shows that $s_1 - s_2 \in \varphi(R)$. Next, observe

$$s_1 s_2 = \varphi(r_1)\varphi(r_2) = \varphi(r_1 r_2),$$

showing that $s_1 s_2 \in \varphi(R)$.Finally, since $1_R \in R$, that tells us that $\varphi(1_R) = 1_S$ since φ is a ring homomorphism. Thus, $\varphi(R)$ is a subring of S.

- To prove $R/ker(\varphi) \cong \varphi(R)$, I will construct the following map: $\phi : R/ker(\varphi) \to \varphi(R)$ where $\phi(r + ker(\varphi)) = \varphi(r)$ and prove that ϕ is an isomorphism. First, to see the ϕ is well defined, let $r_1, r_2 \in R$ such that $r_1 - r_2 \in ker(\varphi)$ and observe

$$\phi(r_1) = \phi(r_2 + (r_1 - r_2)) = \phi(r_2) + \phi(r_1 - r_2) = \phi(r_2) + 0 = \phi(r_2).$$

Next, let $r_1 + I$, $r_2 + I \in R/I$. Due to the fact that φ is a homomorphism, we have the following:

$$\begin{aligned}\phi(r_1 + I + r_2 + I) &= \phi(r_1 + r_2 + I) = \varphi(r_1 + r_2) = \varphi(r_1) + \varphi(r_2)\\ &= \phi(r_1 + I) + \phi(r_2 + I)\\ \phi((r_1 + I)(r_2 + I)) &= \phi(r_1 r_2 + r_1 I + r_2 I + I) = \phi(r_1 r_2 + I + I + I)\\ &= \phi(r_1 r_2 + I) = \varphi(r_1 r_2) = \varphi(r_1)\varphi(r_2)\\ &= \phi(r_1 + I)\phi(r_2 + I)\\ \phi(1 + I) &= \varphi(1) = 1\end{aligned}$$

The above calculations shows that ϕ is a homomorphism. Lastly, I must show that ϕ is a bijections. For injectivity, I will equivalently prove that $ker(\phi) = \{0\}$: Assume that $r + ker(\varphi) \in ker(\phi)$. This implies that $\phi(r + I) = \varphi(r) = 0$ which implies that $r \in ker(\varphi)$, stating that the $ker(\phi) = \{0\}$. For subjectivity, let $s \in \varphi(R)$. This allows us to say there exists an $r \in R$ such that $\varphi(r) = s \Rightarrow \phi(r + ker(\varphi)) = s$ which tells us that $s \in \phi(R)$, implying that ϕ is surjective. Therefore, we have that ϕ is an isomorphism.

□

With the the above ring theory in mind, we can then define what it means to be a field.

Definition 2.4 (see [21]). *A* ***field*** *is a ring, F, with the following conditions satisfied:*

1. $0_F \neq 1_F$,

2. *Every non-zero element has a multiplicative inverse, aka $\forall a \neq 0 \in F, \exists b \in F$ such that $ab = ba = 1$. We usually denote a's multiplicative inverse as a^{-1} for convenience.*

3. *$(F, \times)$ is abelian.*

Definition 2.5 (see [21]). *The* ***characteristic*** *of a field $\mathbb{F}$, denoted $ch(\mathbb{F})$, is defined to be the smallest possible integer p such that $p \times 1_{\mathbb{F}} = 0$, if such a p exists. If no such p exists, then it is defined to be zero.*

For MPKC, we usually work with finite fields. A **finite field** is a field whose elements are finite. For clarity on the definition of a field as well as the structure of finite fields, see the example below.

Example 2.1. *We define $\mathbb{F}_q$ as follows for prime q:*

- $\mathbb{F}_q = \{0, 1, 2, \ldots, q-1\}$,

- $a + b = a + b \ (mod\ q)$ *for all* $a, b \in \mathbb{F}_q$,

- $ab = ab \ (mod\ q)$ *for all* $a, b \in \mathbb{F}_q$,

Now to show $\mathbb{F}_q$ is a field. Also show that $\mathbb{F}_q$ has characteristic q.

Now that we have a space to work in, let's discuss what we work with in multivariate cryptography. We need a structure to be able to use polynomials within these fields. These polynomials that live in finite fields are crucial as they are the public key of many multivariate crypto schemes.

Definition 2.6 (see [2]). *Let $\mathbb{F}$ be a field. If $a_n, a_{n-1}, \ldots, a_1, a_0 \in \mathbb{F}$ where n is a non-negative integer, then any expression of the form*

$$a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$$

*is called a **polynomial over** $\mathbb{F}$, with **indeterminate** x and **coefficients** $\{a_i\}$. If $p(x) = \sum_{i=0}^{n} a_n x^n$, we say $p(x)$ has **degree** n, written as deg$(p(x)) = n$. If the leading coefficient of $p(x)$ is 1, then we say $p(x)$ is a **monic polynomial**. The set of all polynomials with coefficients in $\mathbb{F}$ is denoted by $\mathbb{F}[x]$.*

We call $\mathbb{F}[x]$ a **univariate polynomial ring** due to it having a single indeterminate. However, this structure is very limited in its application to cryptography as we usually use a single indeterminate for a single "part" or "section" of the intended message to be encrypted. This is explained further in 2.3. Thus, we need rings that have many indeterminates. The definition for such a structure is as follows:

Definition 2.7 (see [35]). *If R is a ring, then the **multivariate polynomial ring** in n variables $X = \{X_1, \ldots, X_n\}$ over R is the set of all finite expressions of the forms of $p(X) = \sum a_{i_1 \ldots i_n} X_1^{i_1} \cdots X_n^{i_n}$, where $a_{i_1 \ldots i_n} \in R$ and the i_j are nonnegative integers. Such polynomials are added and multiplied in the usual way. The multivariate polynomial ring is denoted $R[X]$ or $R[X_1, \ldots, X_n]$. The **total degree** of a monomial term, $a_{i_1 \ldots i_n} X_1^{i_1} \cdots X_n^{i_n}$, is defined to be $\sum_{k=1}^{n} i_k$. The **total degree** of a polynomial, $p(X)$, is defined to be the maximum of the total degrees of p's nonzero monomials.*

Definition 2.8 (see [2]). *Let $\mathbb{F}$ be a field, and let $f(x)$ be a fixed polynomial over $\mathbb{F}$. If $a(x), b(x) \in \mathbb{F}[x]$, then we say that $a(x)$ and $b(x)$ are **congruent modulo** $f(x)$, written as $a(x) \equiv b(x) \pmod{f(x)}$ if and only if $f(x) \mid (a(x) - b(x))$. The set $\{b(x) \in \mathbb{F}[x] \mid a(x) \equiv b(x) \pmod{f(x)}\}$ is called a **congruence class** of*

$a(x)$, and is denoted as $[a(x)]$.

The set of all congruence classes modulo $f(x)$ will be denoted $\mathbb{F}[x]/\langle f(x)\rangle$.

Being able to reduce a function to its congruence class representative is imperative in MPKC. The following theorem is a great tool for this process.

Theorem 2.2 (see [2]). ***Fermat**: If p is a prime number, then for any integer a we have*

$$a^p \equiv a \pmod{p}$$

Proof. If $p \mid a$, then it is trivial that $a^p \equiv a \equiv 0 \pmod{p}$. Otherwise $gcd(a,p) = 1$ and Euler's theorem states that $a^{\phi(p)} \equiv 1 \pmod{p} \Rightarrow a^p \equiv a \pmod{p}$ since $\phi(p) = p-1$. □

An example of when you have two different polynomials representing the same function defined over $\mathbb{F}[x]$ is provided below.

Example 2.2. *Let $f(x) = x$ and $g(x) = x^5$ where $f, g \in \mathbb{F}_5[x]$. Observe that, for all $c \in \mathbb{F}_q$, we have $g(c) = c^5 \equiv c \pmod{5}$ by Fermat's Theorem. Thus, both function are members of the same equivalence class over $\mathbb{F}_5$. It is common to use the representative of the residual class when working with them. In this case, $f(x) = x$ would be the representative as it is the "simplest" version of all members.*

Another, less intuitive, example is as follows:

Example 2.3. *Let $f(x) = x^5 - 2x + 1$ and $g(x) = 4x + 1$ where $f, g \in \mathbb{F}_5$. Let's follow a similar line of logic as example 2.2. Let $c \in \mathbb{F}_5$. Observe*

$$f(c) = c^5 - 2c + 1 \equiv c - 2c + 1 \equiv -c + 1 \equiv 4c + 1 \pmod{5}.$$

With the ability to discuss polynomials over finite fields, there is still an issue that needs to be addressed. It is often necessary to be able to factor a polynomial completely over our field. However, it is not always the case that it is possible to do so. The polynomial may be **irreducible** over $\mathbb{F}$.

Definition 2.9 (see [2]). *A non-constant polynomial over $\mathbb{F}$ is considered **irreducible over** $\mathbb{F}$ if it cannot be factored into a polynomial of lower degree. It is considered **reducible** if such a factorization exists.*

Thus, we often need to construct fields that contain all roots of our polynomials. The first step in this process involves the extension of fields:

Definition 2.10 (see [2]). *Let $\mathbb{K}$ and $\mathbb{F}$ be fields. If $\mathbb{F}$ is a subset of $\mathbb{K}$ and has the operations induced by $\mathbb{K}$, then $\mathbb{F}$ is a subfield of $\mathbb{K}$, and $\mathbb{K}$ is called an **extension field** of $\mathbb{F}$.*

Now that we know we can have larger fields containing what we started with along with the same operations, let's discuss a way to create them. Assume we start with a field $\mathbb{F}$. Now, say we have a polynomial $f(x) \in \mathbb{F}[x]$ that had degree greater than or equal to 2 and there exists a root of $f(x)$ that is not in $\mathbb{F}$, lets call it u. So, for clarity, $f(u) = 0$ but $u \notin \mathbb{F}$. So, we want to generate the smallest field that includes all of $\mathbb{F}$ and the root, u. We define such a set as $\mathbb{F}(u)$. Now, is $\mathbb{F}(u)$ an extension field over $\mathbb{F}$? Of course. Below is an example of such a construction.

Example 2.4. *Let $\mathbb{Q}$ be the field of rational numbers. Let $f(x) = x^2 + 1$. Observe that this polynomial has two roots, but neither of them are in $\mathbb{Q}$. Let $i = \sqrt{-1}$. The field $\mathbb{Q}(i) = \{a + bi\}$ is the smallest field containing $\mathbb{Q}$ as a subfield as well as all roots to $f(x)$. Note that the second root, $-i$, is in $\mathbb{Q}(i)$ with $a = 0$ and $b = -1$.*

Within big field schemes, we like to build up a construction that relates constructing extension fields as well as working with polynomial fields. This is can be accomplished with the following terminology and lemma.

Definition 2.11 (see [2]). *Let $\mathbb{K}$ be an extension of $\mathbb{F}$ and let $u \in \mathbb{K}$. u is said to be **algebraic over** $\mathbb{F}$ if there exists a nonzero polynomial $f(x) \in \mathbb{F}[x]$ such that $f(u) = 0$. If u does not satisfy any nonzero polynomial in $\mathbb{F}[x]$, aka $f(u) \neq 0$ for any $f(x) \in \mathbb{F}[x]$, then u is said to be **transcendental over** $\mathbb{F}$.*

Definition 2.12 (see [2]). *Let $\mathbb{K}$ be an extension field of $\mathbb{F}$, and let u be an algebraic element of $\mathbb{K}$. The monic polynomial $p(x)$ of minimal degree in $\mathbb{F}[x]$ such that $p(u) = 0$ is called the* ***minimal polynomial*** *of u over $\mathbb{F}$. The degree of the minimal polynomial of u over $\mathbb{F}$ is called the* ***degree*** *of u over $\mathbb{F}$.*

Now we have all the necessary terminology. What follows is how we can construct an extension field that contains polynomials.

Lemma 2.1 (see [2]). *Let $\mathbb{K}$ be an extension of $\mathbb{F}$, and let $u \in \mathbb{K}$. If u is algebraic over $\mathbb{F}$, then $\mathbb{F}(u) \cong \mathbb{F}[x]/\langle p(x)\rangle$, where $p(x)$ is the minimal polynomial of u over $\mathbb{F}$.*

Let $\mathbb{K} = \mathbb{F}[x]/\langle p(x)\rangle$ where $degree(p(x)) = n$. We say that $\mathbb{K}$ is a **degree n extension of** $\mathbb{F}$. While working within these structures in MPKC, it is often beneficial to view them through their vector space representations. What follows is the terminology and theorems necessary to justify this viewpoint.

Proposition 2.1 (see [2]). *If $\mathbb{K}$ is an extension field of $\mathbb{F}$, then $\mathbb{K}$ is a vector space over $\mathbb{F}$.*

This structure allows us to use the degree of the minimal polynomial to work with a defined degree vector space. This is proven in the following proposition.

Proposition 2.2 (see [2]). *Let $\mathbb{K}$ be an extension field of $\mathbb{F}$, and let $u \in \mathbb{K}$ be an element algebraic over $\mathbb{F}$. If the minimal polynomial of u over $\mathbb{F}$, $p(x)$, has degree n, then $\mathbb{F}(u) \cong \mathbb{F}[x]/\langle p(x)\rangle$ is an n-dimensional vector space over $\mathbb{F}$.*

Proof. Let $p(x) = \sum_{i=0}^{n} a_i x^i$ be the minimal polynomial of u over $\mathbb{F}$. Let $\mathcal{B} = \{1, u, u^2, \ldots, u^{n-1}\}$. Note that, in this context, each element of $\mathcal{B}$ is an n-dimensional vector. Observe that $\mathbb{F}(u) \cong \mathbb{F}[x]/\langle p(x)\rangle$, and that each coset of $\mathbb{F}[x]/\langle p(x)\rangle$ contains a unique representative of degree less than n. Thus, this isomorphism tells us that each element of $\mathbb{F}(u)$ can be represented uniquely as $\sum_{i=0}^{n} a_i u^i$. This tells us that $\mathcal{B}$ spans $\mathbb{F}(u)$, and the uniqueness of each representation gives us that $\mathcal{B}$ is a linearly independent set of vectors. □

Here is a chance to look at the benefits of this kind of structure. Let $\mathbb{K}$ be an n-dimensional extension of $\mathbb{F}_q$ via $\mathbb{F}_q/\langle p(x)\rangle$, where $p(x)$ is a minimal polynomial for $u \in \mathbb{K}$ of degree n. Let $a(x), b(x) \in \mathbb{K}$. We can define addition of such elements by simple polynomial addition. However, multiplication of such elements is where the beauty of such a structure shines. The multiplication of $a(x)$ and $b(x)$ is defined by computing $a(x)b(x)$, then finding the representative of degree less than n. This process is demonstrated in the following example.

Example 2.5. *Let $\mathbb{K}$ be a degree 3 extension of $\mathbb{F}_3 = GF(3) = \{0,1,2\}$, via $\mathbb{F}_3/\langle x^3 + 2x+1\rangle$. Let $a(x) = x^2+x+1$ and $b(x) = 2x^2+x+2$. Observe that $x^3+2x+1 = 0 \Rightarrow x^3 = -2x-1 = x+2$. Thus, we have the following computations:*

$$a(x)+b(x) = x^2+x+1+2x^2+x+2 = 3x^2+2x+3 = 2x$$

and

$$\begin{aligned} a(x)b(x) &= (x^2+x+1)(2x^2+x+2) = 2x^4+x^3+2x^2+2x^3+x^2+2x+2x^2+x+2 \\ &= 2x^4+3x^3+5x^2+3x+2 = 2x^4+2x^2+2 = 2x(x^3)+2x^2+2 \\ &= 2x(x+2)+2x^2+2 = 2x^2+4x+2x^2+2 = 4x^2+4x+2 \\ &= x^2+x+2 \end{aligned}$$

Along with this, we have another very powerful tool that allows us to quickly compute very large degrees of polynomials.

Proposition 2.3 (see [21]). *Let $\mathbb{F}$ be a field of characteristic p. Then, for any $a, b \in \mathbb{F}$,*

$$(a+b)^p = a^p + b^p \quad \text{and} \quad (ab)^p = a^p b^p.$$

*Put another way, the p^th power map defined by $\varphi(a) = a^p$ is an injective field homomorphism from $\mathbb{F}$ to $\mathbb{F}$. This map is called the **Frobenius Endomorhpism**.*

Proof. Note that we can apply the binomial theorem to any commutative ring, thus a finite field is valid. Observe, by applying the binomial theorem, we have

$$(a+b)^p = a^p + \binom{p}{1}a^{p-1}b + \cdots + \binom{p}{i}a^{n-i}b^i + \cdots + b^p.$$

Let's take a closer look at $\binom{p}{i}$ for prime p and integer $1 \leq i \leq p-1$. Observe

$$\binom{p}{i} = \frac{p!}{i!(p-i)!}.$$

Notice that the factor of p does not cancel out, since $gcd(p, \alpha) = 1 \forall 1 \leq \alpha \leq p-1$ since p is prime. Thus, $\binom{p}{i} \cong 0 \pmod{p}$. This gives us $(a+b)^p = a^p + b^p$. Note that the result can be extended to powers of p, giving us

$$(a+b)^{p^i} = a^{p^i} + b^{p^i} \text{ for any } i \in \mathbb{Z}$$

□

2.2 Gröbner Basis

A good understanding of Gröbner bases is necessary when approaching the topic of MPKC. They are the basis for a standard method of attacking such schemes and knowledge of these objects helps one understand how to construct schemes that do not possess such a vulnerability and these algorithms are an active field of research today. While these algorithms continue to evolve and improve, cryptographers and cryptanalysts need to be aware of these developments and respond accordingly.

2.2.1 Gröbner Basis Basics

Before we dive into known algorithms for computing Gröbner bases, let's begin with understanding the fundamentals necessary to find them. First, note that we work in multivariate polynomial rings in MPKC. In order to work within this structure, it is important to understand the different options for placing an ordering on elements within $\mathbb{F}[X_1, \ldots, X_n]$.

Definition 2.13 (see [35])**.** *Let $\mathbb{F}[X_1, \ldots, X_n]$ be a multivariate polynomial ring.*

- *In the **lexicographical ordering**, the terms are listed in the same order in which the monomials (ignoring constants) would occur in a dictionary if they were words in an alphabet consisting of $X_1, \ldots, X_n$ letters.*

  Example 2.6. *Let $f(x_1, x_2, x_3, x_4, x_5) = 3x_4x_2^3x_1^2 + x_1x_3^2x_2 + x_5x_2x_1 + x_2 + x_5x_2$. The polynomial is rewritten as*

  $$f(x_1, \ldots, x_5) = x_1x_2x_3^2 + 3x_1^2x_2^3x_4 + x_1x_2x_5 + x_2 + x_2x_5$$

  under the lexicographical ordering.

- *In the **degree-lexicographical ordering** the monomials are listed from highest to lowest total degree, and the terms with a fixed total degree are listed in lexicographical ordering.*

  Example 2.7. *The polynomial described in example 2.6 is rewritten as*

  $$f(x_1, \ldots, x_5) = 3x_1^2x_2^3x_4 + x_1x_2x_3^2 + x_1x_2x_5 + x_2x_5 + x_2$$

  under the degree-lexicographical ordering.

For our purposes, we will be using *degree-lexicographical ordering.* It is imperative that an ordering is chosen and used consistently throughout the process of Gröbner Basis algorithms. Failure to hold true to an ordering will cause such algorithms to produce inaccurate results and possibly never terminate.

There are many terms associated with polynomials within multivariate polynomial rings for these algorithms. All such terminology is defined here for reference. For these definitions, $\mathbb{F}[X_1, \ldots, X_n]$ is a multivariate polynomial ring and $f(X) = \sum a_{i_1 \ldots i_n} X_1^{i_1} \ldots X_n^{i_n}$ is an element of it.

Definition 2.14 (see [22]). *For $\mathbb{F}[X_1, \dots, X_n]$, the set $T[X_1, \dots, X_n]$ is the set of all terms in these variables.*

Example 2.8. *Let $\mathbb{F}_2[x_1, x_2, x_3]$ be a multivariate polynomial ring over $GF(2)$. Then,*

$$T[x_1, x_2, x_3] = \{x_1x_2x_3, x_1x_2, x_1x_3, x_2x_3, x_1, x_2, x_3\}.$$

Note that $x_i^j = x_i \ \forall j \in \mathbb{Z}$ since $x_i \in GF(2)$.

Definition 2.15 (see [22]). *The set $M(f) = \{a_{i_1 \dots i_n} X_1^{i_1} \cdots X_n^{i_n} \mid a_{i_1 \dots i_n} \neq 0\}$ is the set of monomials of a given polynomial within $\mathbb{F}[X]$.*

Example 2.9. *Let $f \in \mathbb{F}_3[x_1, x_2, x_3]$ such that $f(X) = 2x_1x_2x_3 + x_1x_2 + 2x_2x_3 + 2x_3$. Then, $M(f) = \{2x_1x_2x_3, x_1x_2, 2x_2x_3, 2x_3\}$.*

Definition 2.16 (see [22]). *The set $T(f)$ is the set of **terms** of f, $T(f) = \{X_1^{i_1} \dots X_n^{i_n} \mid a_{i_1 \dots i_n} \neq 0\}$. Note that $T(f) \subseteq T[X_1, \dots, X_n]$, where $f \in \mathbb{F}[X_1, \dots, X_n]$.*

Example 2.10. *For the polynomial, f, defined in example 2.9:*

$$T(f) = \{x_1x_2x_3, x_1x_2, x_2, x_3\}.$$

Definition 2.17 (see [22]). *For $f \in \mathbb{F}[X_1, \dots, X_n]$, we define the following*

- *The **head term** of f, denoted as $HT(f)$, is $HT(f) = max(T(f))$.*
- *The **head monomial** of f, denoted as $HM(f)$, is $HM(f) = max(M(f))$.*
- *The **head coefficient** of f, denoted as $HC(f)$, is $HC(f) = \frac{HM(f)}{HT(f)}$.*

Example 2.11. *For the polynomial, f, defined in example 2.9:*

- $HT(f) = x_1x_2x_3$
- $HM(f) = 2x_1x_2x_3$

- $HC(f) = \frac{HM(f)}{HT(f)} = \frac{2x_1x_2x_3}{x_1x_2x_3} = 2$

Definition 2.18 (see [22]). *Let $F \subseteq \mathbb{F}[X_1, \dots, X_n]$. We can easily extend the definitions of 2.17 in the following manner:*

- $HT(F) = \{HT(f) \mid f \in F\}$
- $HM(F) = \{HM(f) \mid f \in F\}$
- $T(F) = \bigcup T(f)$ *such that $f \in F$. The definition of $T(F)$ given here differs from the definition given by Faugére in [22]. However, this is a more accurate representation of what is intended.*

With all the terminology under our belt, we can continue to move towards understanding what Gröbner Bases are and how to find them. Before we can understand what a Gröbner Basis is, we need to understand an essential process, polynomial modulation.

Definition 2.19 (see [35]). *We say that f **reduces to** h **modulo** g **in one step** if $a_{\mathbf{i}}X^{\mathbf{i}} \in M(f)$ is divisible by $HM(g)$ and*

$$h = f - \frac{a_{\mathbf{i}}X^{\mathbf{i}}}{HM(g)}g,$$

we denote this as $f \xrightarrow{g} h$. In the important case that $HM(g) \mid HM(f)$, we have

$$h = f - \frac{HM(f)}{HM(g)}g.$$

Example 2.12. *Let $f, g \in \mathbb{F}_3[x_1, x_2, x_3]$ where $f(x_1, x_2, x_3) = x_1^2x_2x_3 + 2x_2x_3 + x_3$ and $g(x_1, x_2, x_3) = x_2x_3 + x_1$. Note we can say the following:*

$$h_1 = f - \frac{x_1^2x_2x_3}{HM(g)}g = (x_1^2x_2x_3 + 2x_2x_3 + x_3) - \frac{x_1^2x_2x_3}{x_2x_3}(x_2x_3 + x_1) = 2x_2x_3 + 2x_1 + x_3$$

$$h_2 = f - \frac{2x_2x_3}{HM(g)}g = (x_1^2x_2x_3 + 2x_2x_3 + x_3) - \frac{2x_2x_3}{x_2x_3}(x_2x_3 + x_1) = x_1^2x_2x_3 + x_1 + x_3$$

From example 2.12, we can see that $f \xrightarrow{g} h_1$ and $f \xrightarrow{g} h_2$ where $h_1 \neq h_2$. However, if you continue the modulation, you can see that $f \xrightarrow{g} h_1 \xrightarrow{g} x_3$ and $f \xrightarrow{g} h_2 \xrightarrow{g} x_3$. This is concerning. The choice of which monomial within f you modulate has an effect on the result. However, example 2.12 seems to indicate that you will always get the same result if you continue the modulation as far as possible. Unfortunately, this is simply untrue.

We just finished exploring the concept of modulating a single polynomial by another multiple times in a chain. However, there is no restriction preventing us from doing this with more than one polynomial. Observe:

EXAMPLE 2.13. *Let $f, g_1, g_2, g_3 \in \mathbb{F}_3[x_1, x_2, x_3]$ where*

$$g_1(x_1,x_2,x_3) = x_1x_2 + x_3 \qquad g_2(x_1,x_2,x_3) = x_2x_3 + x_1 \qquad g_3(x_1,x_2,x_3) = x_2$$

$$f(x_1,x_2,x_3) = x_1^2x_2 + x_2x_3^2 + 2x_1x_3 + x_2^2 + x_2$$

Observe that

$$\begin{aligned} f \xrightarrow{g_3} h_1 &= f - \frac{x_1^2x_2}{x_2}(x_2) = x_2x_3^2 + 2x_1x_3 + x_2^2 + x_2 \\ h_1 \xrightarrow{g_3} h_2 &= h_1 - \frac{x_2x_3^2}{x_2}(x_2) = 2x_1x_3 + x_2^2 + x_2 \\ h_2 \xrightarrow{g_3} h_3 &= h_2 - \frac{x_2^2}{x_2}(x_2) = 2x_1x_3 + x_2 \\ h_3 \xrightarrow{g_3} h_4 &= h_3 - \frac{x_2}{x_2}(x_2) = 2x_1x_3 \end{aligned}$$

Thus, $f \xrightarrow{g_3} h_1 \xrightarrow{g_3} h_2 \xrightarrow{g_3} h_3 \xrightarrow{g_3} h_4 = 2x_1x_3$. We cannot modulate anymore since neither $HM(g_1)$, $HM(g_2)$, or $HM(g_3)$ divide into any monomial of h_4. However, let us try a different ordering on the modulation:

$$\begin{aligned} f \xrightarrow{g_1} h_1' &= f - \frac{x_1^2x_2}{x_1x_2}(x_1x_2 + x_3) = x_2x_3^2 + x_1x_3 + x_2^2 + x_2 \\ h_1' \xrightarrow{g_2} h_2' &= h_1' - \frac{x_2x_3^2}{x_2x_3}(x_2x_3 + x_1) = x_2^2 + x_2 \\ h_2' \xrightarrow{g_3} h_3' &= h_2' - \frac{x_2^2}{x_2}(x_2) = x_2 \\ h_3' \xrightarrow{g_3} h_4' &= h_3' - \frac{x_2}{x_2}(x_2) = 0 \end{aligned}$$

Thus, $f \xrightarrow{g_1} h_1' \xrightarrow{g_2} h_2' \xrightarrow{g_3} h_3' \xrightarrow{g_3} h_4' = 0$.

Note that the order in which we chose polynomials to perform the modulation had a drastic impact on the result. Of course, we would always like to be able to choose an ordering in such a way to modulate to 0, if possible. Since we were able to modulate to 0, it shows that a combination of the polynomials used in modulation exists to form f, specifically $f = x_1(g_1) + x_3(g_2) + (x_2+1)(g_3)$. Due to this possibility of confusion, $F = \{g_1, g_2, g_3\}$ is not a good basis for the ideal generated by g_1, g_2, and g_3. We define the process of modulating a polynomial by a set of polynomials as

Definition 2.20 (see [35]). *Let* $F = \{g_1, \dots, g_l\} \subset \mathbb{F}[X_1, \dots, X_n]$ *and let* $f \in \mathbb{F}[X_1, \dots, X_n]$. *We say that* f ***reduces to*** h ***modulo the set of polynomials*** F *if we have a sequence of polynomials beginning with* $h_0 = f$ *and ending with* $h_k = h$ *such that* h_j *reduces to* h_{j+1} *modulo some* $g \in F$ *in one step,* $j = 0, 1, \dots, k-1$.

This issue of different results from modulating a polynomial by a set of polynomials leads us into one of the foundational concepts in computational algebra, finding a good choice of basis polynomials for an ideal.

Definition 2.21 (see [35]). *Let* $G = \{g_1, \dots, g_l\} \subset \mathbb{F}[X] = \mathbb{F}[X_1, \dots, X_n]$ *be a finite set of polynomials in* n *variables over a field* $\mathbb{F}$. *Let* I *be the ideal of* $\mathbb{F}[X]$ *that they generate,* $I = \langle g_1, \dots, g_l \rangle$. *We say that* G *is a* ***Gröbner Basis*** *for the ideal* I *if every nonzero* $f \in I$ *has a leading term that is divisible by the leading term of at least one of the generators,* $g_i \in \{g_1, \dots, g_l\}$.

It is worth pointing out the the term Gröbner **Basis** can be misleading. Indeed, a Gröbner Basis is a collection of functions that generate an ideal, in the sense that every element of the ideal can be represented as a linear combination of Gröbner Basis elements. However, it is not a **Basis** in the sense of vector spaces, i.e. where such a linear combination would be unique. Before we try and use these objects, it would be nice to know if there exists a Gröbner Basis for every ideal, I. To do so, we need to introduce the concept of general polynomial division.

Definition 2.22 (see [21]). ***General Polynomial Division***: *Fix a monomial ordering on* $\mathbb{F}[X] = \mathbb{F}[X_1, \ldots, X_n]$ *and suppose* $g_1, \ldots, g_l$ *is a collection of nonzero polynomials in* $\mathbb{F}[X]$*. If* $f \in \mathbb{F}[X]$*, start with a set of quotients* $q_1, \ldots, q_l$ *and a remainder* r *initially equal to 0 and successively test if the leading term of the dividend,* f*, is divisible by the leading terms of the divisors,* $g_1, \ldots, g_l$*, in that order. Then,*

1. *If* $HT(g_i) \mid HM(f) \Rightarrow HM(f) = a_i HM(g_i)$*, set* $q_i + = a_i$ *and replace* $f := f - a_i g_i$*, and reiterate the entire process.*

2. *If the leading term of* f *is not divisible by any of the leading terms of* $g_1, \ldots, g_l$*, set* $r + = HM(f)$*, replace* $f := f - HM(f)$*, and reiterate the entire process.*

This process terminates when the dividend is 0 and results in a set of quotients, $q_1, \ldots, q_l$*, and a remainder,* r*, with*

$$f = (\sum_{i=1}^{l} q_i g_i) + r.$$

Each $q_i g_i$ *has degree less than or equal to the degree of* f *and the remainder* r *has the property that no nonzero term in* r *is divisible by any of the leading terms of* $g_1, \ldots, g_l$*. This is by construction, since only monomials with this property are added to* r*.*

Theorem 2.3 (see [21]). *Fix a monomial ordering on* $\mathbb{F}[X]$ *and suppose* $G = \{g_1, \ldots, g_l\}$ *is a Gröbner Basis for a non-zero ideal,* I *in* $\mathbb{F}[X]$*. Then,*

1. *Every polynomial* $f \in \mathbb{F}[X]$ *can be written uniquely in the form of*

$$f = f_1 + r$$

where $f_1 \in I$ and no nonzero monomial term of the "remainder" r is divisible by any of the leading terms of the polynomials within G, i.e. $\forall\ g_i \in G,\ HM(g_i) \nmid r_j,\ \forall\ r_j \in M(r)$.

2. *Both f_1 and r can be computed by general polynomial division by $g_1, \ldots, g_l$ and are independent of the order in which these polynomials are used in the division.*

3. *The remainder r provides a unique representative for the coset of f in the quotient ring $\mathbb{F}[X]/I$. In particular, $f \in I$ if and only if $r = 0$.*

Proposition 2.4 (see [21])**.** *Fix a monomial ordering on $\mathbb{F}[X]$ and let I be a nonzero ideal in $\mathbb{F}[X]$. Define $ILT(I) = \langle HM(f) \mid f \in I\rangle$ to be the **ideal of leading terms**, i.e. the ideal generated by the leading terms of all elements in the ideal.*

1. *If $g_1, \ldots, g_l$ are any elements of I such that $ILT(I) = \langle HM(g_1), \ldots, HM(g_l)\rangle$, then $\{g_1, \ldots, g_l\}$ is a Gröbner basis for I.*

2. *The ideal, I, has a Gröbner basis.*

The definition of a Gröbner Basis is lacking in its ability to instruct how to find one. Once found, they have many applications, including breaking MPKC schemes. So, how does one find/compute, given a collection of functions, a Gröbner Basis of the ideal generated by those functions? That question was first answered by Dr. Buchberger in his Ph.D. thesis, [9]. The first theorem presented is used in the proof of Buchberger's Theorem.

Theorem 2.4 (see [35])**.** *$F \subset \mathbb{F}[X]$ is a Gröbner Basis for an ideal, I, if and only if every $f \in I$ reduces to 0 modulo F.*

Proof. ($\Rightarrow$) Assume that F is a Gröbner basis for an ideal, I. Let $f \in I$ be nonzero. (If it were 0, then it is trivial that it reduces to 0 modulo F.) Note that there exists a $g_i \in F$ such that $HM(g) \mid HM(f)$, thus we can modulate f, $f \xrightarrow{g_i} h_1$. Observe that $f > h_1$ according the ordering placed on $\mathbb{F}[X_1, \dots, X_n]$. It is easy to see that $h_1 \in I$. Thus, we can repeat the process to look at $h_1 \xrightarrow{g_k} h_2$ where $g_k \in F$ and $f > h_1 > h_2$. Since the result of each modulation stays in I and continues to decrease down the imposed ordering, we will end with 0.

($\Leftarrow$) Assume every $f \in I$ reduces to 0 modulo F. Let $f \in I$ be nonzero. Since f reduces to 0 modulo F, there exists a finite sequence of polynomials in F, $\{g_1, \dots, g_k\}$, such that $f \xrightarrow{g_1} h_1 \xrightarrow{g_2} h_2 \xrightarrow{g_3} \cdots \xrightarrow{g_{k-1}} h_{k-1} \xrightarrow{g_k} h_k = 0$. Thus, for some $g_i \in g_1, \dots, g_k \subset F$, $HM(g_i) \mid HM(f)$. Therefore, F is a Gröbner Basis for I. □

Definition 2.23 (see [35]). *The **S-Polynomial** of two polynomials $f, g \in \mathbb{F}[X]$ is*

$$S(f,g) = \frac{L}{HM(f)}f - \frac{L}{HM(g)}g$$

where L is the least common multiple of the leading terms of f and g. This can be computed (for algorithmic purposes) as

$$L = lcm(\alpha, \beta) \prod_{i=1}^{n} X_i^{max(\alpha_i, \beta_i)}$$

where $HM(f) = \alpha \prod_{i=1}^{n} X_i^{\alpha_i}$ and $HM(f) = \beta \prod_{i=1}^{n} X_i^{\beta_i}$.

Theorem 2.5 (see [35]). ***Buchberger**: $F \subset \mathbb{F}[X]$ is a Gröbner Basis for I if and only if $S(g_i, g_j)$ reduces to 0 modulo F for every $g_i, g_j \in F$.*

This theorem allows us to develop an algorithm in which to construct Gröbner bases, see Appendix C. However, the Buchberger algorithm is not efficient. Thus, Faugére improved upon this concept in creating the F4 algorithm, [22]. For reference, I also include his pseudo code in Appendix C. There are a few definitions necessary to understand his algorithm.

Definition 2.24 (see [22]). *A critical pair of two polynomials (f_i, f_j) is an ele-*

ment of $T^2 \times R[X] \times T \times R[X]$*,* $Pair(f_i, f_j) := (lcm_{i,j}, t_i, f_i, t_j, f_j)$ *such that*

$$lcm(Pair(f_i, f_j)) = lcm_{i,j} = HT(t_i f_i) = HT(t_j f_j) = lcm(HT(f_i), HT(f_j))$$

Definition 2.25 (see [22])**.** *We say that the degree of the critical pair* $p_{i,j} = Pair(f_i, f_j)$*, denoted* $deg(p_{i,j})$*, is* $deg(lcm_{i,j})$*. We define two projections* $Left(p_{i,j}) := (t_i, f_i)$ *and* $Right(p_{i,j}) := (t_j, f_j)$*. If* $(t, p) \in T \times R[X]$*, then we note* $mult((t, p))$ *the evaluated product* $t * p$*.*

2.2.2 Gröbner Basis Attacks

Now, we have the terminology necessary to look at Faugére's F4 algorithm. I have constructed the F4 algorithm as well as Buchberger's algorithm in Magma, see [7]. This code is provided in Appendix D for reference. Now that the algorithms for computing Gröbner basis are understood, one can see how they can be used in an attack on MPKC systems.

One of the main attacks that MPKC systems are susceptible to is a Gröbner basis attack through using the F4 and F5 algorithms developed by Faugére. This attack uses a GB algorithm to solve the following system:

$$p_1 = y_1, \ldots, p_n = y_n$$

where p_i are the public key polynomials and y_i is the cipher text that was intercepted. A main process within these algorithms is the search for combinations of public polynomials, $\sum g_i p_i$ where $g_i \in \mathbb{F}[x_1, \ldots, x_n]$, where the degree of the summands, $g_i p_i$, are equal and the resulting degree, $\sum g_i p_i$, is less than expected. This occurs when there is cancellation of highest degree terms. This can occur trivially, but the key to this calculation is when this reduction is non-trivial. Examples of trivial cancellations are as follows:

$$p_i^h p_j^h - p_j^h p_i^h \qquad and \qquad ((p_i^h)^{q-1} - 1) p_i^h = 0.$$

Non-trivial reductions are ones do not occur from the equations above. An interesting phenomenon is that once the first non-trivial cancellation occurs, the algorithm will terminate shortly after. This phenomenon has been supported by extensive experimentation and the algorithm will always encounter this cancellation before terminating. Thus, an understanding of when this "degree fall" is crucial in understanding the complexity of the Gröbner Basis attacks. A formal definition is given below.

Definition 2.26 (see [18]). *Let $\mathbb{K} = \mathbb{F}[x_1, \dots, x_n]/\langle x_1^q - x_1, \dots, x_n^q - x_n\rangle$ be the algebra of functions over $\mathbb{F}_q^n$. Let $p_1, \dots, p_n$ be a set of quadratic polynomials in $\mathbb{K}$. Denote $\mathbb{K}_{\leq k}$ to be the subspace of $\mathbb{K}$ consisting of functions representable by a polynomial of degree less than or equal to k.*

For all j we have a natural map $\psi_j : \mathbb{K}_{\leq j}^n \to \mathbb{K}_{\leq j+2}$ given by

$$\psi_j(a_1, \dots, a_n) = \sum_i a_i p_i,$$

where

$$\mathbb{K}_{\leq j}^n = \mathbb{K}_{\leq j} \times \cdots \times \mathbb{K}_{\leq j}.$$

If at least one of the a_i has a degree j but $\sum_i a_i p_i$ has degree less than $j + 2$, we say that a "degree fall" occurs. Obviously we can have trivial degree falls of the form

$$p_j p_i + (-p_i)p_j = 0 \qquad or \qquad (p_i^{q-1} - 1)p_i = 0.$$

The ***degree of regularity*** *of the set $\{p_1, \dots, p_n\}$ is the smallest degree at which a non-trivial degree fall occurs.*

2.3 Introduction to Cryptography

It is very common to humanize the definition of a cryptosystem with names such as Alice, Bob, and Cathy. For, in the field of cryptography, every situation we study involves the transfer of information between Alice and Bob while Cathy attempts to undermine the security of their transmissions.

The field of cryptography can be split into two main fields of study, **symmetrical** and **asymmetrical** cryptography. The difference between these structures lie in their construction. A **symmetrical** cryptosystem uses the same private information to perform encryption and decryption. Thus, for two individuals like Alice and Bob to communicate they would need to have a private key agreed upon. This can be done by meeting before communication occurs or performing the Diffie-Hellman key exchange, see [15]. An example of the latter will be walked through later in this section. However, for modern implementations, it is very difficult to set up a predetermined meeting before secure communication can occur. Also, the ability to verify the identity of a user of a scheme is becoming more necessary, and the Diffie-Hellman key exchange does not provide the ability to do so.

Thus, another class of cryptosystems were introduced in the late 1970's, **asymmetrical** cryptosystems. Simply, these are cryptosystems that do not depend on both parties having access to the private key information used to construct the scheme. Say Alice would like others to be able to communicate with her securely. She can then construct an asymmetrical cryptosystem. She never discloses her private key, and publishes a **public key**. This public key allows anyone to communicate with her by passing their message through it and sending her the encrypted message. This construction has more applications than just encryption and decryption, which I will explain in more detail later in this section. One of the first cryptosystems of this type is the well known RSA, [58].

In this section, I will go through a basic construction of both a symmetrical and asymmetrical cryptosystem. I will then walk through their strengths and weaknesses. Afterwards, I will discuss why the movement away from current implementations of RSA is necessary because of the development of polynomial time algorithms for quantum computers for factoring and solving the discrete log problem by Shor, [62]. Further, I will walk through one of the first constructions built that is considered *post-quantum* since the "hard" problem it is based on cannot be broken by known quantum algorithms. This scheme, C^*, by Tsutomu Matsumoto and Hideki Imai, see [37], is considered to be one of the first within Multivariate Public Key Cryptography (MPKC).

2.3.1 Symmetrical Cryptography

To begin, lets define common terminology that is used widely throughout the crypo community.

Definition 2.27. *The **plain text** is a "message" in its original form before it is run through a cryptosystem. This may be a message like "attack" or "retreat", but it may also be a random string, which can be used as a key for another cryptosystem. The **cipher text** is the result of running the plain text through a cryptosystem. Hopefully, the meaning is obfuscated and prevents interception of the message bearing fruit.*

Definition 2.28. ***Encryption** is the process of taking a plain text and producing a cipher text. **Decryption** is the process of taking a cipher text and computing its corresponding plaintext.*

Definition 2.29. *The **secret key** is the information necessary to compute $f(\mathcal{P}) = \mathcal{C}$ and $f^{-1}(\mathcal{C}) = \mathcal{P}$. When discussing security, one assumes that an outsider, Cathy, does not have access to the secret key.*

The following is an toy example of a symmetrical encryption system, the Hill Cipher: Let each letter in the alphabet be represented by its numerical equivalent modulo 26:

$$A : 0 \pmod{26}, B : 1 \pmod{26}, \ldots, Z : 25 \pmod{26}$$

Let the secret key be a randomly chosen, invertible 2×2 matrix modulo 26:

$$A = \begin{pmatrix} 1 & 2 \\ 3 & 5 \end{pmatrix} A^{-1} = \begin{pmatrix} 21 & 2 \\ 3 & 25 \end{pmatrix}$$

Encryption: Let the intended message be: HELP. This correlates to the plain text of $7, 4, 11, 15$. Construct the plain text into two 2×1 vectors:

$$\begin{pmatrix} 7 \\ 4 \end{pmatrix} \quad \begin{pmatrix} 11 \\ 15 \end{pmatrix}.$$

Then, apply the key matrix, A to each vector, while working modulo 26:

$$\begin{pmatrix} 1 & 2 \\ 3 & 5 \end{pmatrix} \cdot \begin{pmatrix} 7 \\ 4 \end{pmatrix} = \begin{pmatrix} 15 \\ 15 \end{pmatrix}$$

$$\begin{pmatrix} 1 & 2 \\ 3 & 5 \end{pmatrix} \cdot \begin{pmatrix} 11 \\ 15 \end{pmatrix} = \begin{pmatrix} 15 \\ 4 \end{pmatrix}$$

Finally, rearrange the resulting vectors to get the cipher text: $15, 15, 15, 4$.

Decryption: To decrypt the cipher text, simply apply the inverse key, A^{-1}, to the cipher text after breaking it up to two 2×1 vectors. Observe:

$$\begin{pmatrix} 21 & 2 \\ 3 & 25 \end{pmatrix} \cdot \begin{pmatrix} 15 \\ 15 \end{pmatrix} = \begin{pmatrix} 7 \\ 4 \end{pmatrix}$$

$$\begin{pmatrix} 21 & 2 \\ 3 & 25 \end{pmatrix} \cdot \begin{pmatrix} 15 \\ 4 \end{pmatrix} = \begin{pmatrix} 11 \\ 15 \end{pmatrix}$$

This gives us a plain text of $7, 4, 11, 15$ or HELP, the intended message. This is an example of symmetric cryptography since the secret key $\{A, A^{-1}\}$ was used for encryption and decryption.

2.3.2 Diffie-Hellman Key Exchange

As discussed earlier, there is a need for keys to be exchanged before secure communication to take place in symmetric cryptography. Thus, a method for secure key transfer was constructed by Whitfield Diffie and Martin E. Hellman in 1976, coined the "Diffie-Hellman Key Exchange", see [15]. A key point to understand is that encryption and decryption are not processes associated with this process. The Diffie-Hellman key exchange is a method which allows two users to generate a shared key securly. This process is described below, followed by a toy example for clarification.

Alice and Bob wish to generate a shared secret. Cathy is attempting to undermine their future communications, thus wishes to be able to determine this shared secret, if possible, and is monitoring their communication. Unless specified otherwise, all data transmitted between Alice and Bob is able to be seen by Cathy. This allows a good framework for discussing the security of the scheme by allowing Cathy to know everything possible, except "secret keys", as is the norm for judging security of schemes in cryptography.

Alice and Bob agree to work in the world of modulo p, where p is prime, and use an agreed upon primitive root (an element which generates the multiplicative group of the integers modulo p), g of modulo p. To generate a shared secret, both Alice and Bob choose values a and b respectively, where $1 < a, b < p$. Alice sends $g^a \pmod{p}$ to Bob while Bob sends $g^b \pmod{p}$ to Alice. Alice is able to compute $g^{ab} \pmod{p}$ since she has her secret key, a, and knows $g^b \pmod{p}$ from Bob. bob is also able to compute $g^{ab} \pmod{p}$ since he has his secret key, b, and knows $g^a \pmod{p}$ from Alice. Thus, they have generated a shared secret, $g^{ab} \pmod{p}$, which can then be used as a key for future encryption/decryption using symmetrical cryptography.

Is this shared secret secure? Cathy knows the following: $g, p, g^a \pmod{p}$, and $g^b \pmod{p}$. Can she determine $g^{ab} \pmod{p}$ from that information? No! This problem is also known as the discrete log problem and there is no current algorithm to solve this problem in general.

Toy Example:

Let $p = 23$ and $g = 5$. Let Alice's and Bob's secrets be $a = 6$ and $b = 15$ respectively. Alice sends Bob $5^6 \pmod{23} = 8$ and Bob sends Alice $5^{15} \pmod{23} = 19$. Then, Alice computes $19^6 \pmod{23} = 2$ and Bob computes $8^{15} \pmod{23} = 2$. Thus, both Alice and Bob have generated the shared secret of 2 (mod 23).

2.3.3 Asymmetrical Cryptography

Here, we will discuss the basics of an asymmetrical cryptosystem, commonly referred to as a public key cryptosystem. As mentioned in section 2.3.1, it is necessary for both parties to have access to a secret key in order to use a symmetrical cryptosystem. Also, there is the vulnerability of the ”man-in-the-middle” attack on the Diffie-Hellman key exchange.To have a system that did not depend on a shared secret key while avoiding the vulnerability of the Diffie-Hellan key exchange, public key cryptography was created.

A general description of how a public key cryptosystem is easy to understand. Say Alice wishes to allow anyone to securely communicate with her. Thus, she published a *public key.* This provides Bob, or anyone else, a system to feed their plain text through to generate a corresponding cipher text. Then, Bob can send this cipher text back to Alice. She uses her *private key*, information used to generate the public key but kept secret, to decrypt Bob's message.

This structure is easy to understand and see it's applications, but very difficult to construct. The difficulty lies in the structure's security. Notice that Cathy, someone wishing to be able to undermine the scheme and be able to understand Bob's messages, is able to see the public key. This may not be true, but in determining public key security, we assume an attacker has all structural information about the scheme, except the private key. This provides Cathy many different ways to break the scheme. One possible way is for Cathy to generate all possible plain text and cipher text pairs. Thus, if she were to intercept a cipher text sent from Bob to Alice, she would be able to identify its corresponding plain text. There are other ways to break public key cryptosystems, which will be addressed later.

The structural cornerstone of a public key cryptosystem is its *trap-door function.* The difficulty in creating public key cryptosystems is identifying these useful trap-door functions.

Definition 2.30. *Let $f : \mathcal{P} \to \mathcal{C}$ be a function from the plain text space, $\mathcal{P}$, to the cipher text space, $\mathcal{C}$. Such a function is considered to be a* ***trap-door*** *function if $f(p)$ for any $p \in \mathcal{P}$ is "easy" to compute and $f^{-1}(c)$ is "hard". In other words, by knowing f, it is easy for anyone to compute a plain text's corresponding cipher text. However, being able to find the inverse of f, is "hard" without knowing the* ***private key****. Thus, the individual with the private key can easily decrypt a cipher text, since the private key allows them to quickly find f^{-1}, but anyone without the private key is unable to find $f^{-1}(c)$ in a reasonable amount of time.*

The best way to understand a public key cryptosystem is to work with one. The example we will be using is one of the most well known schemes, RSA [58]. The description given is not the standardized structure used today. However, this will allow you to understand the basics of a public key cryptosystem, the underlying structure of RSA, and why RSA is secure. First, I define Euler's phi-function since it is necessary for the construction, then I give a detailed description of RSA.

Definition 2.31 (see [35])**.** *Let n be a positive integer. The* ***Euler phi-function,*** *$\varphi(n)$, is defined to be the number of nonnegative integers b less than n which are* *relatively prime to n.*

$$\varphi(n) = \left|\{0 \le b < n \mid gcd(b,n) = 1\}\right|$$

Proposition 2.5 (see [36])**.** *If $gcd(a,n) = 1$, then $a^{\varphi(n)} \equiv 1 \pmod{n}$.*

Cryptosystem 2.1 (RSA)**.** *Two large distinct primes, on the magnitude of 100-decimal digits, p and q are chosen and kept secret. Then, one computes $n = pq$ and randomly chooses e, where $1 < e < \varphi(n)$ and $gcd(\varphi(n), e) = 1$. (Note: $\varphi(n)$ is Euler's phi-function) Next, the user computes the the multiplicative inverse of $e \pmod{\varphi(n)}$, which we will call d.*

- ***Private Key**: (p,q,d)*
- ***Public Key**: (n,e)*
- ***Enciphering Function**: $f : \mathbb{Z}_n \to \mathbb{Z}_n$ where $f(\mathcal{P}) = P^e \pmod{n}$*
- ***Deciphering Function**: $f^{-1}(\mathcal{C}) = C^d \pmod{n}$*

Lets discuss some specific details of this system. First of all, it should be noted that the plain text space, $\mathcal{P}$, should only contains elements that are relatively prime to n, but that is not the case. It is very unlikely that plain text elements would fail this condition since n is the product of 2 very large primes, thus the probability for decryption failure is very low. Next,it should be clear that f^{-1} does indeed give you a cipher text's corresponding plain text:

Proof. Let $n = pq$ where p and q are distinct primes. $\varphi(n)$ is Euler's phi function of n. Let e be chosen between 1 and $\varphi(n)$ such that it is relatively prime to $\varphi(n)$. Let d be the multiplicative inverse of $e \pmod{\varphi(n)}$, i.e. $de \equiv ed \equiv 1 \pmod{\varphi(n)}$. Let $c = p^e \bmod n$ be a cipher text that corresponds to the plain text p. Observe:

$$c^d = (p^e)^d = p^{ed} = p^{a\varphi(n)+1} = p^{a\varphi(n)}p \equiv 1 * p \pmod{n} \equiv p \pmod{n}$$

□

The security of this scheme is based on the difficulty of factoring n. Note, that the main element that allows quick decryption is d, the multiplicative inverse of $e \pmod{\varphi(n)}$. If an attacker were able to compute this quantity, they would be able to decrypt as quickly as our user. So, why does knowing d depend upon knowing the factorization of n? Note that we are looking for a multiplicative inverse $\pmod{\varphi(n)}$. Thus, one needs to be able to compute $\varphi(n)$. Computing this quantity quickly requires you to know the factorization of $n = pq$. If you know this, then $\varphi(n) = (p-1)(q-1)$. It is noted in [58] that factoring such large numbers (for example, n being 200 digits) can take 3.8×10^9 years. For clarity, a toy example of a basic RSA implementation is provided below.

Toy Example: Bob wishes to generate a public key so that Alice can communicate with him in a secure manner. Thus, Bob chooses to use an RSA implementation with the following **secret** parameters: $p = 3$, $q = 11$, $\varphi(n) = 20$, and $d = 3$. With this, he generates the following public key: $P_A = \{n = 33, e = 7\}$. Note that the exponent is chosen such that $gcd(e, \varphi(n)) = 1$. Alice chooses to send the message $M = 14$ to Bob, after feeding it through the RSA encryption of $C = M^e \pmod{n}$, resulting in $C = 20 \pmod{33}$. Bob receives the cipher text, computes $20^3 \pmod{33}$ in order to decrypt the cipher text to receive the message $M = 14$.

Encryption and decryption are not the only concepts that are relevant to asymmetrical cryptography. Digital signatures is another necessity of modern technology that utilizes public key cryptosystems. This can be viewed as an electronic extension of one's physical signature, as on a check or lease. These signatures are intended to be unique to the individual and prove that it was you who authorized said transaction. How can this be done electronically? Public key cryptosystems allow for this to be done with a small change to the standard encryption protocol.

For our description, assume that Alice and Bob are both using an RSA public key system. Thus, they each have published an encryption key, (n_A, e_A) and

(n_B, e_B), for Alice and Bob respectively. These keys describe the following encryption functions: $f_A(\mathcal{P}) = \mathcal{P}^{e_A} \pmod{n_A}$ and $f_B(\mathcal{P}) = \mathcal{P}^{e_B} \pmod{n_B}$. For example, say Bob wishes to send a message, c, to Alice and prove to Alice it was him who sent the message. It would not suffice for him to just encrypt the message using Alice's public key, as anyone who can see her public key could do that. So, Bob does not just send his message, he sends his message concatenated with his signature: a hash of the original message. For details on hash functions, see Appendix B.

For our example, lets say he wishes to send a message m and sign it with $h(m)$, where $h(x)$ is a shared hash function between Alice and Bob (it is not necessary that it is kept secret from Cathy). So, Bob is going to send $\{m, h(x)\}$. Again, he cannot just feed both of these through f_A as many people could know the hash function and could do the same, pretending to be Bob. Thus, Bob is going to send the following: $\{f_A(m), f_A(f_B^{-1}(h(m)))\}$. So, when Alice receives this message, claiming to be from Bob, she feeds the encrypted message through her inverse function and sees the following: $f_A^{-1}(\{f_A(m), f_A(f_B^{-1}(h(x)))\}) = \{m, f_B^{-1}(h(m))\}$. The end (or beginning, doesn't matter) of the message is unintelligible to Alice. Since Bob claimed to send this message, Bob would be the only one to truly know f_B^{-1}. Knowing this, Alice feeds the unreadable ending of the message through Bob's published f_B. Remember, she can do this as everyone has access to f_B. She can then see the following: $f_B(f_B^{-1}(h(m))) = h(m)$, the hash of the intended message.

Alice can then feed her intended decrypted message, m, through the shared hash function, $h(x)$, and confirm in fact that they are the same. This provides Alice with a few reassurances. First, it is clear that Bob sent the message since he is the only one aware of f_B^{-1}. Second, it is clear that m was the intended message and not tampered with. This is due to the fact that $h(m)$ in the signature is equal to the hash applied to the decrypted message sent.

RSA was an amazing development in cryptography. However, the difficult problem it is based has been broken through Shor's algorithm, [62], using quantum computers. Thus, once usable quantum computers become a reality, all data and communications secured using RSA will become vulnerable. Thus, a movement for Post-Quantum Cryptography, PQC, began a few decades ago. This movement is focusing on developing cryptography that is not based upon problems that we know quantum computers have an advantage on, including factorization as well as the discrete log problem. There are a few main fields of PQC: multivariate, lattice, and code-based. In this document, we will be exploring recent developments within multivariate post quantum cryptography that I have published. Before we dive into those, one should be exposed to the first multivariate scheme, C^*. By understanding C^*'s structure, it will be mush easier to understand the more complex schemes discussed in later chapters.

2.3.4 Multivariate Post Quantum Cryptography

In [37], Matsumoto and Imai developed one of the first multivariate post quantum encryption scheme. This scheme is considered to be post quantum since the difficult problem its security is based on is solving a system of quadratic equations over a finite field, which is considered to be an NP-Hard problem in the general case CITE. One of the most interesting aspects of this scheme is the use of Frobenius automorphisms in order to achieve high power computations with ease when generating the public key. Let's look at the construction:

Let $\mathbb{F}_q$ be a finite field with characteristic 2. Let $p(x)$ be an irreducible polynomial of degree n over $\mathbb{F}_q$. Let $\mathbb{K}$ be the n degree field extension described as $\mathbb{F}_q[x]/\langle p(x)\rangle \cong \mathbb{K}$. Let $\phi : \mathbb{K} \to \mathbb{F}_q^n$ be the standard vector space isomorphism defined as $\phi(a_0 + a_1x + \cdots + a_{n-1}x^{n-1}) = (a_0, a_1, \ldots, a_{n-1})$. Let $T, U : \mathbb{F}_q^n \to \mathbb{F}_q^n$ be two affine transformations. Choose θ such that $0 < \theta < n$ and $gcd(q^\theta + 1, q^\theta - 1) = 1$. Let the **core map** $f \in \mathbb{K}$ be defined as $f(X) = X^{q^\theta+1}$. This construction generates f in such a way that it is a bijection and it has an inverse, $f^{-1}(X) = X^\alpha$ where α is an integer such that $\alpha(1+q^\theta) \equiv 1 \pmod{q^n - 1}$. The public key is generated by $P = T \circ \phi^{-1} \circ f \circ \phi \circ U = T \circ F \circ U$ where $F = \phi^{-1} \circ f \circ \phi$. This can be viewed in the following diagram:

$$\begin{tikzcd}
 & \mathbb{K} \arrow[r, "f"] & \mathbb{K} \arrow[d, "\phi^{-1}"] & \\
\mathbb{F}_q^n \arrow[r, "U"] & \mathbb{F}_q^n \arrow[u, "\phi"] \arrow[r, "F"] & \mathbb{F}_q^n \arrow[r, "T"] & \mathbb{F}_q^n
\end{tikzcd}$$

Private Key: The private key is the two affine transformations, T and U.

Public Key: The functions $(P_1, P_2, \ldots, P_n) = T \circ \phi^{-1} \circ f \circ \phi \circ U$ where $P_i \in \mathbb{F}_q[X] = \mathbb{F}_q[x_1, x_2, \ldots, x_n]$

Encryption: Let the plain text be $(x_1, \ldots, x_n)$ where $x_i \in \mathbb{F}_q$. The cipher text $(y_1, \ldots, y_n)$ is computed by plugging in the plain text into the public key equations, i.e. $y_i = P_i(x_1, \ldots, x_n)$.

Decryption: Given $(y_1, \ldots, y_n)$, the holder of the private key can recover the plain text by the following computations:

- Compute $(u_1, \ldots, u_n) = U^{-1}(y_1, \ldots, y_n)$
- Compute $(u'_1, \ldots, u'_n) = \phi^{-1} \circ f^{-1} \circ \phi \ (u_1, \ldots, u_n)$
- Compute $(x_1, \ldots, x_n) = T^{-1}(u'_1, \ldots, u'_n)$

Let's take a look at a simple toy example. Such parameters do not provide security for a scheme, but will allow the reader to understand the method of constructing the public key as well as the encryption and decryption process.

Example 2.14. *Let $\mathbb{F}_2$ be a finite field of 2 elements and let $\mathbb{K}$ be a degree 3 extension so that $\mathbb{F}_2[X]/\langle x^3 + x + 1\rangle \cong \mathbb{K}$. Let $\phi : \mathbb{K} \to \mathbb{F}_2^3$ be a vector space isomorphism. Let T, U, and their corresponding inverses be the following affine transformations* *over $\mathbb{F}_2^3$:*

$$T = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, T^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, U = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix}, U^{-1} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

Let $f(X) = X^{2^2+1} = X^5$. Thus, we know that $f^{-1}(X) = X^3$ since $3(1+2^2) \equiv 1 \pmod{2^3-1}$. Let's begin constructing the public key. Let β be the root to the irreducable polynomial $p(x) = x^3 + x + 1$. Note that we are looking to construct an element of $\mathbb{F}_2^3$ while looking at $\phi \circ f \circ \phi^{-1}$. We can let $\mathcal{B} = \{1, \beta, \beta^2\}$ be the basis for vector space representation of $\mathbb{K}$. Note that we will be using the following basis element representations:

$$1 \mapsto \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \beta \mapsto \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \beta^2 \mapsto \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

First, we compute $U \cdot (x_1, x_2, x_3)^\intercal$, which we will call $\bar{u}$, for a general plain text (x_1, x_2, x_3):

$$U \cdot (x_1, x_2, x_3)^\intercal = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} x_1 + x_2 + x_3 \\ x_1 \\ x_1 + x_2 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \bar{u}$$

Next, we need to compute $\phi \circ f \circ \phi^{-1}(\bar{u})$. The first step is to see that $\phi^{-1}(\bar{u}) = u_1 + u_2\beta + u_3\beta^2$. We then feed this into f:

$$f(u_1 + u_2\beta + u_3\beta^2) = (u_1 + u_2\beta + u_3\beta^2)^5 = (u_1 + u_2\beta + u_3\beta^2)^4(u_1 + u_2\beta + u_3\beta^2)^1$$

We split the exponentiation into powers of $q = 2$ in order to take advantage of the frobenius automorphisms mentioned in section 2.1. Thus, we get the following, *remembering that $u^i = u$ for all positive i since we are in $GF(2)$:*

$$\begin{aligned}
&(u_1 + u_2\beta + u_3\beta^2)^4(u_1 + u_2\beta + u_3\beta^2)^1 = (u_1^4 + u_2^4\beta^4 + u_3^4\beta^8)(u_1 + u_2\beta + u_3\beta^2)\\
&= (u_1 + u_2\beta^4 + u_3\beta^8)(u_1 + u_2\beta + u_3\beta^2)\\
&= u_1^2 + u_1u_2\beta + u_1u_3\beta^2 + u_1u_2\beta^4 + u_2^2\beta^5 + u_2u_3\beta^6 + u_1u_3\beta^8 + u_2u_3\beta^9 + u_3^2\beta^{10}\\
&= u_1 + u_1u_2\beta + u_1u_3\beta^2 + u_1u_2\beta^4 + u_2\beta^5 + u_2u_3\beta^6 + u_1u_3\beta^8 + u_2u_3\beta^9 + u_3\beta^{10}
\end{aligned} \tag{2.1}$$

Remember that β is the root to the irreducible polynomial $p(x) = x^3 + x + 1$. Thus, we can get representations for higher degrees of β in terms of $\{1, \beta, \beta^2\}$. This can be done in a few different ways. I am going to highlight the method of finding a squaring matrix to reduce the number of calculations. Due to the structure of the vectors that are mapped from the basis elements, you can generate the squaring matrix from left to right by placing the vector that corresponds to the first, second, and third basis element squared respectively. So,

$$1^2 = 1 \mapsto \begin{bmatrix}1\\0\\0\end{bmatrix}, (\beta)^2 = \beta^2 \mapsto \begin{bmatrix}0\\0\\1\end{bmatrix}, (\beta^2)^2 = \beta^4 = \beta^2 + \beta \mapsto \begin{bmatrix}0\\1\\1\end{bmatrix}$$

We know that $\beta^4 = \beta^2 + \beta$ by multiplying both sides of $\beta^3 = \beta + 1$, which we know because $p(\beta) = \beta^3 + \beta + 1 = 0$, by β. The technique of multiplying both sides of these equations by β until you reach the desired power of β can be continued until all desired powers are found. However, we will need to do this 10 times until we reach β^{10}, the highest power we need. This process can be shortened by finding the squaring matrix, which will allow us to find $\beta^8 = (\beta^4)^2$ in a single computation. This may not seem to save work in this toy example, but when powers of β get very large, this technique is very efficient in saving computation time. Since we now know β^4, *we can finish constructing the squaring matrix:*

$$SqM = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix}, since\ \beta^2 + \beta = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix}$$

Now that we know the squaring matrix, we will use it to get the vector representations of β^8 and then use the technique of multiplying both sides by β to get β^9 and β^{10} from β^8.

$$\beta^4 = \beta + \beta^2$$

$$\beta^5 = \beta^2 + \beta^3 = \beta^2 + \beta + 1$$

$$\beta^6 = (\beta^3)^2 = SqM \cdot \beta^3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} = 1 + \beta^2$$

$$\beta^8 = (\beta^4)^2 = SqM \cdot \beta^4 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} = \beta$$

$$\beta^9 = \beta^2$$

$$\beta^{10} = \beta^3 = \beta + 1$$

Now that we have all necessary powers of β in terms of our basis elements, we can *then return to equation* 2.1*:*

$$u_1 + u_1u_2\beta + u_1u_3\beta^2 + u_1u_2\beta^4 + u_2\beta^5 + u_2u_3\beta^6 + u_1u_3\beta^8 + u_2u_3\beta^9 + u_3\beta^{10}$$

$$= u_1 + u_1u_2\beta + u_1u_3\beta^2 + u_1u_2(\beta + \beta^2) + u_2(\beta^2 + \beta + 1) + u_2u_3(1 + \beta^2)$$

$$+ u_1u_3(\beta) + u_2u_3(\beta^2) + u_3(\beta + 1)$$

$$= (u_1 + u_2 + u_2u_3 + u_3)(1) + (u_1u_2 + u_1u_2 + u_2 + u_1u_3 + u_3)(\beta)$$

$$+ (u_1u_3 + u_1u_2 + u_2 + u_2u_3 + u_2u_3)(\beta^2)$$

$$= (u_2u_3 + u_1 + u_2 + u_3)(1) + (u_2 + u_1u_3 + u_3)(\beta) + (u_1u_2 + u_1u_3 + u_2)(\beta^2)$$

Then we send the result through ϕ*:*

$$\phi((u_2u_3 + u_1 + u_2 + u_3)(1) + (u_2 + u_1u_3 + u_3)(\beta) + (u_1u_2 + u_1u_3 + u_2)(\beta^2))$$

$$= (u_2u_3 + u_1 + u_2 + u_3)\begin{bmatrix}1\\0\\0\end{bmatrix} + (u_2 + u_1u_3 + u_3)\begin{bmatrix}0\\1\\0\end{bmatrix} + (u_1u_2 + u_1u_3 + u_2)\begin{bmatrix}0\\0\\1\end{bmatrix}$$

$$= \begin{bmatrix}u_2u_3 + u_1 + u_2 + u_3\\ u_2 + u_1u_3 + u_3\\ u_1u_2 + u_1u_3 + u_2\end{bmatrix} = \bar{u}'$$

Finally, we apply T*:*

$$T\bar{u}\prime = \begin{bmatrix}0 & 0 & 1\\ 1 & 0 & 0\\ 0 & 1 & 1\end{bmatrix} \cdot \begin{bmatrix}u_2u_3 + u_1 + u_2 + u_3\\ u_2 + u_1u_3 + u_3\\ u_1u_2 + u_1u_3 + u_2\end{bmatrix} = \begin{bmatrix}u_1u_2 + u_1u_3 + u_2\\ u_2u_3 + u_1 + u_2 + u_3\\ u_2 + u_1u_3 + u_3 + u_1u_2 + u_1u_3 + u_2\end{bmatrix}$$

$$= \begin{bmatrix}u_1u_2 + u_1u_3 + u_2\\ u_2u_3 + u_1 + u_2 + u_3\\ u_3 + u_1u_2\end{bmatrix}$$

Next, substitute the values of x that each u_i represent:

$$
\begin{aligned}
T\bar{u}\prime = \begin{bmatrix} u_1u_2 + u_1u_3 + u_2 \\ u_2u_3 + u_1 + u_2 + u_3 \\ u_3 + u_1u_2 \end{bmatrix} &= \begin{bmatrix} (x_1 + x_2 + x_3)(x_1) + (x_1 + x_2 + x_3)(x_1 + x_2) + (x_1) \\ (x_1)(x_1 + x_2) + (x_1 + x_2 + x_3) + (x_1) + (x_1 + x_2) \\ (x_1 + x_2) + (x_1 + x_2 + x_3)(x_1) \end{bmatrix} \\
&= \begin{bmatrix} (x_1 + x_1x_2 + x_1x_3) + (x_1 + x_1x_2 + x_1x_3) + (x_1x_2 + x_2 + x_2x_3) + (x_1) \\ (x_1 + x_1x_2) + (x_1 + x_2 + x_3) + (x_1) + (x_1 + x_2) \\ (x_1 + x_2) + (x_1 + x_1x_2 + x_1x_3) \end{bmatrix} \\
&= \begin{bmatrix} x_1x_2 + x_2 + x_2x_3 + x_1 \\ x_1x_2 + x_3 \\ x_2 + x_1x_2 + x_1x_3 \end{bmatrix}
\end{aligned}
$$

Thus, the public key is $P(X) = (P_1(X), P_2(X), P_3(X))$ where $P_1(X) = P_1(x_1, x_2, x_3) = x_1x_2 + x_2 + x_2x_3 + x_1$, $P_2(X) = P_2(x_1, x_2, x_3) = x_1x_2 + x_3$, and $P_3(X) = P_1(x_1, x_2, x_3) = x_2 + x_1x_2 + x_1x_3$.

***Encryption**: Say Alice published the public key, $P(X)$. Bob wishes to send the plain text $(1, 0, 1)$, so $(x_1, x_2, x_3) = (1, 0, 1)$. Bob recalls the public key, $P(X)$ and begins to feed his plain text into the system of equations:*

$$
P(1,1,0) = \begin{bmatrix} x_1x_2 + x_2 + x_2x_3 + x_1 \\ x_1x_2 + x_3 \\ x_2 + x_1x_2 + x_1x_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}
$$

Bob then sends the cipher text $(1, 1, 1)$ to Alice.

***Decryption**: Alice is able to decrypt any message received by using her private key information. Let's construct the general decryption algorithm, then use that to decrypt Bob's message. Say Alice receives a cipher text $(y_1, y_2, y_3) = \bar{y}$. First, she* *will apply T^{-1} to $\bar{y}$:*

$$T^{-1}\bar{y} = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} y_2 \\ y_1 + y_3 \\ y_1 \end{bmatrix} = \bar{t} = \begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}$$

Next, Alice feeds this result through $\phi \circ f^{-1} \circ \phi^{-1}$, recalling that $f^{-1} = X^3$:

$$\phi \circ f^{-1} \circ \phi^{-1}(\bar{t}) = \phi \circ f^{-1}(t_1 + t_2\beta + t_3\beta^2)$$

We will need to repeat a similar process that was used during the public key generation to compute $f^{-1}(t_1 + t_2\beta + t_3\beta^2)$:

$$f^{-1}(t_1 + t_2\beta + t_3\beta^2) = (t_1 + t_2\beta + t_3\beta^2)^3$$

$$= (t_1 + t_2\beta + t_3\beta^2)^2(t_1 + t_2\beta + t_3\beta^2)^1$$

$$= (t_1 + t_2\beta^2 + t_3\beta^4)(t_1 + t_2\beta + t_3\beta^2)$$

$$= t_1 + t_1t_2\beta + t_1t_3\beta^2 + t_1t_2\beta^2 + t_2\beta^3 + t_2t_3\beta^4 + t_1t_3\beta^4 + t_2t_3\beta^5 + t_3\beta^6$$

$$= t_1 + t_1t_2\beta + t_1t_3\beta^2 + t_1t_2\beta^2 + t_2(\beta + 1) + t_2t_3(\beta^2 + \beta)$$

$$+ t_1t_3(\beta^2 + \beta) + t_2t_3(\beta^2 + \beta + 1) + t_3(\beta^2 + 1)$$

$$= (t_1 + t_2 + t_2t_3 + t_3)(1) + (t_1t_2 + t_2 + t_2t_3 + t_1t_3 + t_2t_3)(\beta)$$

$$+ (t_1t_3 + t_1t_2 + t_2t_3 + t_1t_3 + t_2t_3 + t_3)(\beta^2)$$

$$= (t_1 + t_2 + t_2t_3 + t_3)(1) + (t_1t_2 + t_2 + t_1t_3)(\beta) + (t_1t_2 + t_3)(\beta^2)$$

Then, we feed the result through ϕ:

$$\phi((t_1 + t_2 + t_2t_3 + t_3)(1) + (t_1t_2 + t_2 + t_1t_3)(\beta) + (t_1t_2 + t_3)(\beta^2)) = \begin{bmatrix} t_1 + t_2 + t_2t_3 + t_3 \\ t_1t_2 + t_2 + t_1t_3 \\ t_1t_2 + t_3 \end{bmatrix} = \bar{t}'$$

Finally, Alice then computes $U^{-1} \cdot \bar{t}' =$*:*

$$= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} t_1 + t_2 + t_2 t_3 + t_3 \\ t_1 t_2 + t_2 + t_1 t_3 \\ t_1 t_2 + t_3 \end{bmatrix} = \begin{bmatrix} t_1 t_2 + t_2 + t_1 t_3 \\ t_2 + t_1 t_3 + t_3 \\ t_1 + t_2 + t_2 t_3 + t_1 t_2 \end{bmatrix}$$

$$= \begin{bmatrix} (y_2)(y_1 + y_3) + (y_1 + y_3) + (y_2)(y_1) \\ (y_1 + y_3) + (y_2)(y_1) + (y_1) \\ (y_2) + (y_1 + y_3) + (y_1 + y_3)(y_1) + (y_2)(y_1 + y_3) \end{bmatrix} = \begin{bmatrix} y_2 y_3 + y_1 + y_3 \\ y_1 y_2 + y_3 \\ y_1 y_2 + y_1 y_3 + y_2 y_3 + y_2 + y_3 \end{bmatrix}$$

Now Alice has a decrypting system. She can proceed to find the corresponding plain text for the received cipher text by plugging in $(y_1, y_2, y_3) = (1, 1, 1)$ *into her decrypting system:*

$$\begin{bmatrix} y_2 y_3 + y_1 + y_3 \\ y_1 y_2 + y_3 \\ y_1 y_2 + y_1 y_3 + y_2 y_3 + y_2 + y_3 \end{bmatrix} = \begin{bmatrix} 1 + 1 + 1 \\ 1 + 1 \\ 1 + 1 + 1 + 1 + 1 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$$

Thus, Alice was able to successfully recover the intended plain text message, $(1, 0, 1)$ *sent by Bob.*

CHAPTER 3
HFE

Shortly after Patarin broke C^*, [47], he constructed a new multivariate scheme called *Hidden Field Equations*, often referred to as HFE [48]. This new construction was designed to resist the method Patarin used in breaking Matsumoto and Imai's scheme. In this chapter, I will give a detailed description of HFE and HFEm with a toy example and current parameter suggestions indicated by the literature, walk through a detailed look at the security of HFE and HFEm through the lens of differential security, and end on the breakthrough by Dr. Smith-Tone and myself in breaking HFEm in [66].

3.1 HFE and HFEm

3.1.1 Scheme Description

Let $\mathbb{F}_q$ be a finite field of size q. Let $\mathbb{K}$ be an n degree extension of $\mathbb{F}_q$, i.e. $\mathbb{K} = \mathbb{F}_q/\langle p(x)\rangle$ where $p(x)$ is an irreducible polynomial of degree n over $\mathbb{F}_q$. Note that we can view $\mathbb{K}$ as a n degree vector space over $\mathbb{F}_q$ by letting $\phi : \mathbb{F}_q^n \to \mathbb{K}$ be the natural $\mathbb{F}_q$ vector space isomorphism. Since a generator of $Gal_{\mathbb{F}_q}(\mathbb{K})$ is the Frobenius automorphism, $x \mapsto x^q$, every monomial map of the form $f(x) = x^{q^i+q^j}$ in $\mathbb{K}$ can be viewed as a vector valued function over $\mathbb{F}_q$ through the following representation: $\phi^{-1} \circ f \circ \phi$. One can then apply this concept repeatedly to see that any vector valued quadratic map on $\mathbb{F}_q^n$ is isomorphic to a sum of such monomials. This structure motivates the following core map, $f \in \mathbb{K}$, for HFE:

$$f(x) = \sum_{\substack{i \le j \\ q^i + q^j \le D}} \alpha_{i,j} x^{q^i + q^j} + \sum_{\substack{i \\ q^i \le D}} \beta_i x^{q^i} + \gamma, \tag{3.1}$$

where the coefficients $\alpha_{i,j}, \beta_i, \gamma \in \mathbb{K}$ and the degree bound D is sufficiently low for efficient inversion, [38]. Let $\mathcal{B}$ be a chosen basis for $\mathbb{K}$, then one can express f in the basis $\mathcal{B}$ as n polynomials of degree 2 with n variables:

$$f(x_1, \ldots, x_n) = \{f_1(x_1, \ldots, x_n), \ldots, f_n(x_1, \ldots, x_n)\} \tag{3.2}$$

where $f_i(\mathcal{X}) \in \mathbb{F}_q[\mathcal{X}]$. Similar to C^*, we will apply two affine transformations, T and U, to hide the choice of basis for $\mathbb{K}$. The public key is generated by $P = T \circ \phi^{-1} \circ f \circ \phi \circ U$. This technique is used throughout MPKC that it is conveniently named the **butterfly construction** and can be easily viewed in the following diagram:

$$\begin{tikzcd}
 & \mathbb{K} \arrow[r, "f"] & \mathbb{K} \arrow[d, "\phi^{-1}"] & \\
\mathbb{F}_q^n \arrow[r, "U"] & \mathbb{F}_q^n \arrow[r, "F"] \arrow[u, "\phi"] & \mathbb{F}_q^n \arrow[r, "T"] & \mathbb{F}_q^n
\end{tikzcd}$$

Private Key: f, T, U, and structure of $\mathbb{K}$

Public Key: $P = T \circ \phi^{-1} \circ f \circ \phi \circ U = (P_1, \ldots, P_n)$, redundancy technique, and $\mathbb{F}_q^n$

Encryption: Feed the plain text through a chosen redundancy technique to get $(x_1, \ldots, x_n)$ as a result where $x_i \in \mathbb{F}_q$. The cipher text $(y_1, \ldots, y_n)$ is computed by plugging in the plain text into the public key equations, i.e. $y_i = P_i(x_1, \ldots, x_n)$.

Decryption: Given $(y_1, \ldots, y_n)$, the holder of the private key can recover the plain text by the following computations:

- Compute $(u_1, \ldots, u_n) = U^{-1}(y_1, \ldots, y_n)$
- Compute $(u_1', \ldots, u_n') = \phi^{-1} \circ f^{-1} \circ \phi \ (u_1, \ldots, u_n)$
- Compute $(x_1, \ldots, x_n) = T^{-1}(u_1', \ldots, u_n')$

- Undo the chosen redundancy technique.

Two methods of redundancy are described in [48], redundancy in or outside the plain text. Redundancy in the plain text requires your chosen message to be of length less than n since there will be redundancy added to obtain length n. For example, say your message is $M = (x_1, \ldots, x_{n-l})$. Then, you would feed $\bar{x} = M \| h(M)$ where $\|$ represents concatenation and $h(M)$ is the first l digits of a hash function applied to M. Hash functions like MD5 or SHS would suffice. For details on these hash functions, see Appendix B.

For redundancy outside the plain text, you apply the hash to the output of the scheme, $y = (y_1, \ldots, y_n)$. Thus, the final cipher text would be $Y = y \| h(x)$ where x is the plain text. If h is a one way collision free hash function, then you are guaranteed to find one solution.

Patarin also notes that a randomly generated f may not be a permutation of $\mathbb{K}$, [48]. He then proceeds to introduce the following theorem, which he coins as the "heart" of this new scheme.

Theorem 3.1 (see [48])**.** *Let $\mathbb{K}$ be a finite field with $|\mathbb{K}| = q^n$ with q and n "not too large" (for example $q \leq 64$ and $n \leq 1024$). Let $f(x)$ be a given polynomial in x in a field $\mathbb{K}$, with a degree D "not too large" (for example, $D \leq 1024$). Let $a \in \mathbb{K}$.* ***Then*** *it is always possible (on a computer) to find all the roots of an equation $f(x) = a$.*

Proof. This is a well known result and a proof can be seen in [38] on pg. 17-26. □

Theroem 3.1 is necessary to show that it is possible for the inversion of f, which is necessary in the decryption process. However, a method in which to find such roots is not described in the theorem. There are a few techniques one can employ to find roots of polynomials over finite fields. Such methods are the Berlekamp algorithm [3], as well as techniques described in [45] and [68]. The main technique used today is the Berlekamp algorithm and thus its limitations are placed on the core maps of our schemes. The main limiting factor is the limit placed on the degree on our core map. If D in equation 3.1 is too large, the computation will be too complex for efficient inversion. A detailed description of the algorithm is given in Appendix A.

Recall that we discussed how public key cryptosystems can be used for signatures. This concept easily applies to HFE. The following adaptations are used when HFE is being used for digital signatures as outlined by Patarin in [48]:

For sake of discussion, let us work within $GF(2)$. Let M be the intended message. Let x and $P(x)$ be the plain text and cipher text, respectfully, of an associated HFE instance, P, where x and $P(x)$ have length $n = 128$. Let h be a collision free hash function with output of $n = 128$, MD5 for example, and $\|$ represent concatenation.

Computing the Signature:

1. Generate a small integer R with no block of numbers with 10000_2 in its expression in base 2.

2. Compute $h(R\|10000\|M)$.

3. Let $y = h(R\|10000\|M)$. Then, with the private key of the HFE instance, we can try to find a plain text x such that $P(x) = y$. If successful, then $R\|x$ will be the signature of M. It is possible that you cannot find such a cipher text x. If so, repeat the loop at step 1 with a new R.

Signature Verification Given M and signature $R\|x$.

1. Separate R and x. This can easily be done since the length of x is fixed to be n.

2. Compute $h(R\|10000\|M)$ and $P(x)$.

3. The signature is valid if $h(R\|10000\|M) = P(x)$.

3.1.2 Scheme Adaptations

While HFE is interesting within its own right, there are some suggested variations of HFE given by Patarin in [48]. In this section, HFEm and HFEp will be introduced. There is the other famous adaptation, HFEv, also known as the "vinegar" adaptation. This variation has gained much attention in the PQC community, thus is deserving of its own chapter.

The first adaptation is HFEm, which stands for "HFE minus", also denoted HFE^{-}. This adaptation uses less public polynomials than is generated when creating the public key. For instance, say $(P_1, \dots, P_m)$ is a public key generated from an instance of an HFE algorithm. It is possible to not publish all of these polynomials. Let a represent the number of polynomials kept secret. Thus, $(P_1, \dots, P_{m-a})$ is published.

For an encryption scheme, a must be kept very small in order to be able to recover the intended plain text from a cipher text. Patarin suggests that it is clearly possible if a is either 1 or 2 in [48]. However, further developments have indicated that a must be larger than 1, since that hidden public equation can be recovered via [5].

For a signature scheme, the number of equations removed is allowed to be much larger. Values such as $a = 1, 2$, or $\frac{m}{2}$ should be both practical and efficient according to Patarin, [48].

The other well known adaptation is HFEp. Read as HFE "plus", this adaptation requires you to, as you may have guessed, add more equations to the public key than those generated. Let $(P_1, \dots, P_m)$ be the public polynomials of $\{x_1, \dots, x_n\}$ indeterminants. You can then generate random quadratic polynomials over said n indeterminants to form $\{Q_1, \dots, Q_k\}$. Then, use a secret affine transformation to mix them up.

For an encryption scheme, k is allowed to be very large. There is a restriction where

$$k + n < \frac{n(n+1)}{2}$$

where k is the number of random polynomials and n is the number of indeterminants. Suggested values given by Patarin in [48] are $k = 1, 2$, or $\frac{n}{2}$.

For a signature scheme, k must be very small. This is due to the fact that, given x, the probability of satisfying the extra k equations is $\frac{1}{2^{km}}$. With both m and k small, the scheme is still efficient as it will take approximately 2^{km} tries to obtain a signature, as mentioned by Patarin in [48].

3.2 Proveable Security

After seeing the construction of HFE, an individual new to the crypto world might think that Patarin discovered a perfect signature and encryption scheme for multivariate public key cryptography and that no further development is needed. This is not the case for HFE or any other scheme. In years past, a scheme's security has been determined by trying to attack it with known attacks. However, this technique leaves much to be desired as it in no way provides guarantees for attacks not yet known or even old attacks with slight adaptations. Also, when a scheme is considered "secure" against an attack, it does not necessarily mean that the attack does not break a scheme. Most of the time, the attack is too computationally complex. This would look like the following: *RSA is secure against Schroeppel's Method of factoring due to the fact that, if n is 200 digits long, it would take a classical computer* 1.2×10^{23} *operations or* 3.8×10^{9} *years to factor n using said method.*

Notice that RSA can be broken using Schroeppel's Method, it would just take an unrealistic amount of time. This does not take into consideration advances in factoring algorithms or technological advances which could speed up the computation time. Thus, there has been a push for “provable” security where one can demonstrate that a scheme resists an attack by proving that such an attack cannot break the scheme, regardless of time. HFE was analyzed in this method to show that it is secure against differential adversaries, given a parameter restriction.

3.2.1 Algebraic Background

There are a few algebraic results that Dr. Smith-Tone and Mr. Daniels use to analyze the differential security of HFE and HFEm, see [13]. These are provided here for reference.

Proposition 3.1 (see [13]). *If A and B are two $m \times n$ matrices, then $rank(A) = rank(B)$ if and only if there exists nonsingular matrices C and D such that $A = CBD$.*

Proof. Let A be an $m \times n$ matrix with rank r. Create a $m \times m$ row operations matrix, P, such that PA is in row echelon form. Then, generate the $n \times n$ column operation matrix, Q, to gather all leading 1's to the first r columns. This gives us PAQ, which is a $m \times n$ matrix with an $r \times r$ identity block in the upper left region. Let $I' = PAQ$. Do a similar process with matrix B, giving us $I' = P'BQ'$. Thus:

$$I' = PAQ = P'BQ' \Rightarrow A = (P^{-1}P')B(Q'Q^{-1})$$

where $P^{-1}P'$ and $Q'Q^{-1}$ are non-singular. □

Definition 3.1 (see [13]). *The **minimal polynomial** of a subspace $V \subseteq \mathbb{K}$ as*

$$\mathcal{M}_V(x) = \prod_{v \in V}(x - v).$$

This is the polynomial of minimal degree of which every element of V is a root. Note that $\mathcal{M}_V(x) = 0$ is an $\mathbb{F}_q$-linear equation and, where V has $\mathbb{F}_q$ dimension d ($|V| = q^d$), then $\mathcal{M}_V(x)$ has degree q^d and has the following form:

$$x^{q^d} + b_{d-1}x^{q^{d-1}} + \cdots + b_1 x^q + b_0 x \qquad \textit{where } b_i \in \mathbb{K}$$

To clarify the statement that $\mathcal{M}_V$ is an $\mathbb{F}_q$ linear map, a proof is given below:

Proof. To be considered an $\mathbb{F}_q$-linear map, the map must be homogeneous and preserve addition. For homogeneity, where $x \in \mathbb{K}$ and $c \in \mathbb{F}_q$:

$$\begin{aligned}
\mathcal{M}_V(cx) &= (cx)^{q^d} + b_{d-1}(cx)^{q^{d-1}} + \cdots + b_1(cx)^q + b_0(cx) \\
&= c^{q^d}x^{q^d} + b_{d-1}c^{q^{d-1}}x^{q^{d-1}} + \cdots + b_1c^qx^q + b_0(cx) \\
&= cx^{q^d} + b_{d-1}cx^{q^{d-1}} + \cdots + b_1cx^q + b_0cx \\
&= c(x^{q^d} + b_{d-1}x^{q^{d-1}} + \cdots + b_1x^q + b_0x) \\
&= c\mathcal{M}_V(x).
\end{aligned}$$

For preservation of addition, for $x, y \in \mathbb{K}$:

$$\begin{aligned}
\mathcal{M}_V(x+y) &= (x+y)^{q^d} + b_{d-1}(x+y)^{q^{d-1}} + \cdots + b_1(x+y)^q + b_0(x+y) \\
&= (x^{q^d} + y^{q^d}) + b_{d-1}(x^{q^{d-1}} + y^{q^{d-1}}) + \cdots + b_1(x^q + y^q) + b_0(x+y) \\
&= [x^{q^d} + b_{d-1}x^{q^{d-1}} + \cdots + b_1x^q + b_0x] + [y^{q^d} + b_{d-1}y^{q^{d-1}} + \cdots + b_1y^q + b_0y] \\
&= \mathcal{M}_V(x) + \mathcal{M}_V(y)
\end{aligned}$$

□

Proposition 3.2 (see [13]). *Let $T : \mathbb{K} \to \mathbb{K}$ be an $\mathbb{F}_q$-linear map. Let $\pi : \mathbb{K} \to \mathbb{K}$ be defined by $\pi x = \mathcal{M}_{ker(T)}(x)$. There exists a non-singular $\mathbb{F}_q$-linear map $\tilde{T} : \mathbb{K} \to \mathbb{K}$ such that $Tx = \tilde{T}\pi x$.*

Proof. As shown above, π is an $\mathbb{F}_q$-linear map. Also, note that $ker(\pi) = ker(T)$ is trivial. Note that π and T are additive homomorphisms, implying both are constant on cosets of the kernel. This allows us to construct the following well-defined function: $\tilde{T}x = T\pi^{-1}(x)$. Observe:

$$\tilde{T}\pi(x) = T\pi^{-1}(\pi x) = T(x + ker(T)) = Tx.$$

□

Finally, Dr. Smith-Tone and Daniels characterized all functions from V to $\mathbb{K}$ (analogous to the coordinate ring $\bar{\mathbb{K}}[x]/\langle\mathcal{M}_V(x)\rangle$):

Proposition 3.3 (see [13]). *Let $\mathcal{F}_V$ be the ring of all functions from the $\mathbb{F}_q$-subspace V of $\mathbb{K}$ to $\mathbb{K}$. Then, $\mathcal{F}_V$ is isomorphic to $\mathbb{K}[x]/\langle\mathcal{M}_V(x)\rangle$.*

Proof. Note that the ring of all functions from $\mathbb{K}$ onto itself is $\mathbb{K}[x]/\langle x^{q^n} - x\rangle$. Let $f, g \in \mathbb{K}[x]/\langle x^{q^n} - x\rangle$ such that $f(v) = g(v)$ for all $v \in V$. Thus, for all $v \in V$, v is a root of $(f-g)(x)$ indicating that $(x-v)$ is a linear factor of $(f-g)(x)$ for all $v \in V$. Therefor, $\mathcal{M}_V(x) \mid (f-g)(x)$. Hence, $\langle\mathcal{M}_V(x)\rangle$ is the ideal of functions which send V to zero. This tells us that $\mathbb{K}[x]/\langle x^{q^n}-x, \mathcal{M}_V(x)\rangle$ is the ring of non-trivial functions from V to $\mathbb{K}$. Since $\mathcal{M}_V(x)$ splits in $\mathbb{K}$, $\mathcal{M}_V(x) \mid x^{q^n} - x$. Finally, since there are $(q^n)^{q^d}$ functions from V, of $\mathbb{F}_q$ dimension d, to $\mathbb{K}$, and $|\mathbb{K}[x]/\langle\mathcal{M}_V(x)\rangle| = (q^n)^{q^d}$, one can see that all functions from V to $\mathbb{K}$ are polynomials. □

3.2.2 Differential Adversary

In [13], Dr. Smith-Tone and Mr. Daniels analyzed the differential security of HFE. To fully appreciate this advancement, it is necessary to understand what a differential is, how an attacker can utilize properties of this computation to undermine the scheme, and where this technique has been a successful attack.

Definition 3.2 (see [13]). *The discrete differential of a field map $f : \mathbb{F}_n^q \to \mathbb{F}_q^n$ is given by:*

$$Df(y, x) = f(x+y) - f(x) - f(y) + f(0).$$

This can be viewed as a normalized difference equation with variable interval.

Example 3.1. *Here we shall see a simple computation of the discrete differential given a field map. Let $f : \mathbb{F}_2^3 \to \mathbb{F}_2^3$ be a field map defined as $f(x) = x^{2^3+1} = x^9$, the C^* core map from example 2.14. Computing the discrete differential of this map,* *we have:*

$$\begin{aligned} Df(y,x) &= f(x+y) - f(x) - f(y) + f(0) \\ &= (x+y)^9 - x^9 - y^9 + 0 \\ &= (x+y)^8(x+y) - x^9 - y^9 \\ &= (x^8+y^8)(x+y) - x^9 - y^9 \\ &= (x^9 + xy^8 + yx^8 + y^9) - x^9 - y^9 \\ &= xy^8 + yx^8 \end{aligned}$$

This calculation seems insignificant, however by looking at specific differential relations, one can exploit them to find structural information of the public key that can be used to undermine the security of the scheme. Specifically, one can look at two different properties related to the differential described in [11]:

Definition 3.3 (see [11])**.** *A general linear* ***differential symmetry*** *is a relation of the form of*

$$Df(Mx,a) + Df(x,Ma) = \Lambda_M Df(a,x),$$

where $M, \Lambda_M : \mathbb{K} \to \mathbb{K}$ are $\mathbb{F}_q$-linear maps.

Definition 3.4 (see [11])**.** *Let $f : \mathbb{F}_q^n \to \mathbb{F}_q^m$ be a function. A* ***differential invariant*** *of f is a subspace $V \subseteq \mathbb{K}$ with the property that there is a subspace $W \subseteq \mathbb{K}$ such that*

$$dim(W) \le dim(V) \quad and \quad \forall\ A\ \in\ Span_{\mathbb{F}_q}(Df_i),\ AV \subseteq W.$$

These are the major properties of the discrete differential that an attacker can utilize to undermine the scheme. As mentioned in [13], attacks such as the linearization equations attack of [48] can be viewed through the lens of a discrete differential as an exploitation of the relation $Df(f(x), f(x)) = 0$. Also, the attacks on balanced Oil-Vinegar, seen in [49] and [33], and the attack on SFLASH in [20] can be seen through the lens of a differential.

3.2.3 Symmetric Security

To protect a scheme against a differential adversary looking to take advantage of a differential symmetry property of the public key, one must guarantee that maps M, Λ_M do not exist such that $Df(Mx, a) + Df(x, Ma) = \Lambda_M Df(a, x)$. This was done in [13] by looking at the structure that M and Λ_M have to take and using a unique graphical approach in analizing a generated system of equations. This technique is used again, with some adaptations, when proving security of HFEv, so a detailed description is provided for future reference.

To determine the possibility of a differential symmetry vulnerability, Mr. Daniels and Dr. Smith-Tone found the conditions necessary for a differential symmetry on an HFE core map

$$f(x) = \sum_{\substack{i \leq j \\ q^i + q^j \leq D}} \alpha_{i,j} x^{q^i + q^j}. \tag{3.3}$$

First, they applied the differential to the HFE core map, 3.3: (The first deduction is for a single quadratic term in the sum. Then, since the sum is finite, we can extrapolate the result to the sum's entirety.)

$$\begin{aligned} Df(y, x) &= f(x+y) - f(x) - f(y) + f(0) \\ &= \alpha_{i,j}(x+y)^{q^i+q^j} - \alpha_{i,j}x^{q^i+q^j} - \alpha_{i,j}y^{q^i+q^j} - 0 \\ &= \alpha_{i,j}(x+y)^{q^i}(x+y)^{q^j} - \alpha_{i,j}x^{q^i+q^j} - \alpha_{i,j}y^{q^i+q^j} \\ &= \alpha_{i,j}(x^{q^i} + y^{q^i})(x^{q^j} + y^{q^j}) - \alpha_{i,j}x^{q^i+q^j} - \alpha_{i,j}y^{q^i+q^j} \\ &= \alpha_{i,j}(x^{q^i+q^j} + y^{q^j}x^{q^i} + y^{q^i}x^{q^j} + y^{q^i+q^j}) - \alpha_{i,j}x^{q^i+q^j} - \alpha_{i,j}y^{q^i+q^j} \\ &= \alpha_{i,j}(y^{q^i}x^{q^j} + y^{q^j}x^{q^i}) \end{aligned} \tag{3.4}$$

Thus,

$$Df(y, x) = \sum_{\substack{i \leq j \\ q^i + q^j \leq D}} \alpha_{i,j}(y^{q^i}x^{q^j} + y^{q^j}x^{q^i}).$$

Since Df is a $\mathbb{K}$-bilinear form, there is a convenient representation for $\mathbb{K}$:

$$x \mapsto \begin{bmatrix} x \\ x^q \\ \vdots \\ x^{q^{n-1}} \end{bmatrix}.$$

With this representation, one can view Df as an $n \times n$ symmetric matrix where the (i,j)th and (j,i)th entries for $i \neq j$ are $\alpha_{i,j}$ and the (i,i)th entries are $2\alpha_{i,j}$. You can see that we are beginning to construct a matrix representation for this differential symmetry property. In order to have this representation, we need a matrix representation for M. Since $M : \mathbb{K} \to \mathbb{K}$ is a $\mathbb{F}_q$ linear map, we can always view $Mx = \sum_{i=0}^{n-1} m_i x^{q^i}$ as the following matrix representation:

$$M = \begin{bmatrix} m_0 & m_1 & \dots & m_{n-1} \\ m_{n-1}^q & m_0^q & \dots & m_{n-2}^q \\ \vdots & \vdots & \ddots & \vdots \\ m_1^{q^{n-1}} & m_2^{q^{n-1}} & \dots & m_0^{q^{n-1}} \end{bmatrix}$$

With this representation for M, we can now view the differential applied to f in the following matrix representation:

$$Df(My,x) + Df(y,Mx) = y(M^{\intercal}Df + DfM)x. \tag{3.5}$$

To simplify it a bit, one can consider applying Λ_M to Df:

$$\Lambda_M Df(y,x) = \sum_{k=0}^{n-1} \lambda_k Df(y,x)^{q^k}.$$

To understand how raising Df to the q^k power affects our representation, observe that for a single quadratic term map f and the result from 3.4:

$$\begin{aligned}
Df(y,x)^{q^k} &= [\alpha_{i,j}(y^{q^i}x^{q^j} + y^{q^j}x^{q^i})]^{q^k} \\
&= \alpha_{i,j}^{q^k}(y^{q^i}x^{q^j} + y^{q^j}x^{q^i})^{q^k} \\
&= \alpha_{i,j}^{q^k}([y^{q^i}x^{q^j}]^{q^k} + [y^{q^j}x^{q^i}]^{q^k}) \\
&= \alpha_{i,j}^{q^k}(y^{q^i q^k}x^{q^j q^k} + y^{q^j q^k}x^{q^i q^k}]) \\
&= \alpha_{i,j}^{q^k}(y^{q^{i+k}}x^{q^{j+k}} + y^{q^{j+k}}x^{q^{i+k}})
\end{aligned}$$

Thus, in general, we can see that

$$Df(y,x)^{q^k} = \sum_{\substack{i\leq j \\ q^i+q^j\leq D}} \alpha_{i,j}^{q^k}(y^{q^{i+k}}x^{q^{j+k}} + y^{q^{j+k}}x^{q^{i+k}}).$$

This allows use to see that the new (i,j)th and (j,i)th entries of Df^{q^k} are $\alpha_{i-k,j-k}^{q^k}$ if $i \neq j$ and the new (i,i)th entries are $(2\alpha_{i-k,j-k})^{q^k}$. One can interpret these results as a shifting the entries of the original Df to the right and down k units and raising all entries to the q^kth power.

With the extra step of applying Λ_M, one can view the differential equation through the following simplified matrix equation, often referred to as the **differential symmetric equation**:

$$M^\top Df + DfM = \Lambda_M Df \tag{3.6}$$

Now that we have a matrix representation of the differential symmetric equation, one can analyze this system to determine if a solution M and Λ_M exist. Initially, this may seem like a daunting task. However, Mr. Daniels and Dr. Smith-Tone employed a graphical approach that simplifies this process and proved the following theorem:

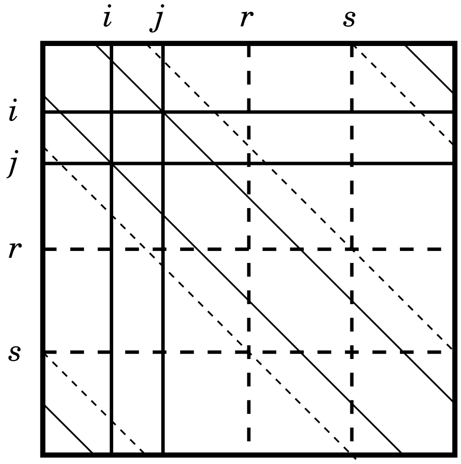
THEOREM 3.2 (see [13]). *Let $f(x)$ be an HFE polynomial (in particular, f is not a monomial function). Suppose that f has the following properties:*

1. No power of q is repeated among the exponents of f, and

2. the difference of the powers of q in each exponent is unique.

Then, f has no nontrivial differential symmetry.

Proof. The proof of the theorem above is dependent on understanding the following figure:



Depicted above is a graphical representation of $M^{\intercal}Df + DfM = \Lambda_M Df$ where $f(x) = \alpha_{i,j}x^{q^i+q^j} + \alpha_{r,s}x^{q^r+q^s}$. Here, horizontal lines represent possible non-zero information from MDf, vertical lines represent possible non-zero information from $M^{\intercal}Df$, and diagonal lines represent possible non-zero information from $\Lambda_M Df$. Imagine the computation of DfM:

$$DfM = \begin{bmatrix} 0 & \dots & \alpha_{i,j} & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \alpha_{i,j} & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & \alpha_{r,s} & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & \alpha_{r,s} & \dots & 0 & \dots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & \dots & 0 & \dots & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} m_0 & m_1 & \dots & m_{n-1} \\ m_{n-1}^q & m_0^q & \dots & m_{n-2}^q \\ \vdots & \vdots & \ddots & \vdots \\ m_1^{q^{n-1}} & m_2^{q^{n-1}} & \dots & m_0^{q^{n-1}} \end{bmatrix}$$

You can see that the ith row of such a calculation would look like:

$$[\alpha_{i,j} m_{-j}^{q^j}\ ,\ \alpha_{i,j} m_{1-j}^{q^j}\ , \dots ,\ \alpha_{i,j} m_{-1-j}^{q^j}] \tag{3.7}$$

and the jth row would be:

$$[\alpha_{i,j} m_{-i}^{q^i}\ ,\ \alpha_{i,j} m_{1-i}^{q^i}\ , \dots ,\ \alpha_{i,j} m_{-1-i}^{q^i}].$$

Now that one can visualize what the left side (LHS) of 3.6 is, imagine what the right hand side (RHS) is. By looking at the provided figure, take the i-th row as an example. It's algebraic representation is given in 3.7. Observe that information for the RHS of 3.6 is represented by diagonal lines. Thus, areas where there is no intersection with our horizontal ith line tell us that $\alpha_{i,j} m_k^{q^j} = 0 \Rightarrow m_k = 0$. This is the breakthrough that allows us to analyze such a large system of equations very quickly. This gives us a massive amount of information about the structure of M, which only has n unknowns. In the analysis, we ignore where intersections of lines occur. This information is too complex to derive any helpful information.

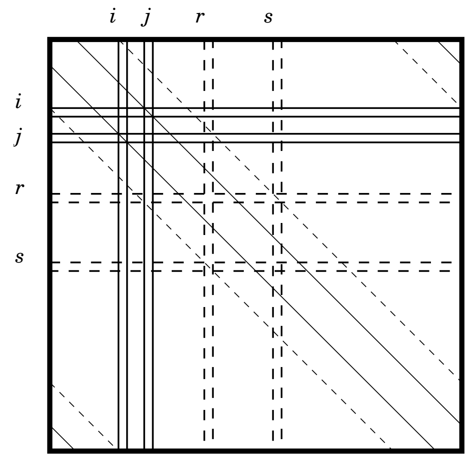
Before moving on, allow me to clarify the restrictions of the theorem. The first, restricting no power of q to be repeated, is in place to prevent the analysis from being too complicated. if a power of q were to be repeated, this would force there to be overlap of rows and columns in our diagram. This would complicate our analysis in the following way: Instead of having the simple equation of $\alpha_{i,j} m_k^{q^j} = 0$ where a horizontal line has no intersection, it would instead be $\sum_l \alpha_l m_{k_l}^{q^j} = 0$ where l is the number of times a power of q is repeated. We would no longer have the simple deduction that $m_k = 0$. This is not to say that by having repetitions of powers of q would make you vulnerable to a differential adversary, it just doesn't fit in our analysis to prove security against such an adversary. The second restriction, the uniqueness of differences of powers of q, is for a similar reason. This would cause overlap on the diagonals, further complicating the analysis.

To formalize our analysis, consider the ith row as indicated in 3.7. For every monomial $\alpha_{r,s} x^{q^r+q^s}$ in f, the $s-r+i$th and $r-s+i$th elements of row i in $\Lambda_M Df$ are the only possible non-zero entries. Thus, for all k not occurring as a power of q or as a difference of powers of q in f plus i, $m_{k-j} = 0$. Due to the restriction that the differences of powers of q are unique in f, along with the fact that $m_{k-t} = 0$ for all t occurring as a power of q, we are able to deduce that $m_i = 0$ for all $i \neq 0$. This makes M a diagonal matrix with coefficients from the base field, thus being a multiplication map. We know that the coefficients come from the base field, $\mathbb{F}_q$, due to Theorem 2 in [63]. This is a trivial differential symmetry since a multiplication map by a scalar from the base field generates a symmetry for every map $g : \mathbb{K} \to \mathbb{K}$.. □

The analysis changes for an HFE^- polynomial. Due to the fact that the analysis for Theorem 3.2 is built from a graphical representation for a system of equations, it would be wise to see the changes in the graphical representation for the differential symmetric equation for an HFE^- polynomial. Due to the nature of the minus modifier being a projection of an HFE polynomial, we can view equation 3.6 as

$$\pi[M^\intercal Df + DfM] = \Lambda_M[Df] \tag{3.8}$$

where π is a projection onto a subspace. In [13], they handled the general case of a co-dimension r projection explicitly. This results in the following graphical representation of 3.8:



with the HFE polynomial $f(x) = \alpha_{i,j}x^{q^i+q^j} + \alpha_{r,s}x^{q^r+q^s}$ and $\pi x = ax + bx^q + x^{q^2}$. Again, horizontal and vertical lines represent possible non-zero entries from the LHS of 3.8 and diagonal lines represent possible non-zero entries in the RHS of 3.8. The analysis of this image lead Dr. Daniel Smith-Tone and Taylor Daniels to prove the following theorem:

Theorem 3.3 (see [13]). *Let $\mathbb{K}$ be a prime extension of $\mathbb{F}_q$ and let $\pi : \mathbb{K} \to \mathbb{K}$ be a co-dimensional r projection. Let $f : \mathbb{K} \to \mathbb{K}$ be a non-trivial HFE polynomial with degree bound $D < q^{n/2}$, let P_f be the multiset of powers of q occuring as exponents of f, and let S_f be the multiset of differences of powers of q in the exponent of each monomial summand of f. Suppose that f has the following properties:*

1. *P_f is a set,*

2. *S_f is a set,*

3. *for all $i \in P_f$ the Lee distance between $(i + S_f) \setminus P_f$ and P_f is at least $r + 1$.*

Then, if $D(\pi \circ f)(My, x) + D(\pi \circ f)(y, Mx) = \Lambda_M Df(y, x)$, then $Mx = m_0x$ for some $m_o \in \mathbb{F}_q$. Thus, $\pi \circ f$ has no non-trivial differential symmetry.

Proof. Due to the effect of T and Proposition 3.2, we can, without loss of generality, assume that $\pi x = \sum_{b=0}^{r} a_b x^{q^b}$ with $a_r = 1$. This gives us the ability to construct the matrix form for $\pi[M^T Df + Df M]$ from the matrix form of $M^T Df + Df M$. When being raised to the power of q, this results in each element of the matrix raised to the power of q and shifted one row down and one column to the right.

Let $\alpha_{i,j} x^{q^i+q^j}$ be a monomial summand of f. Observe that the $(i,k)th$ entry of $\pi[M^T Df + Df M]$ for $k \notin P_f \cup (1+P_f) \cup \cdots \cup (r+P_f) \cup (i+S_f)$ is $m_{k-j}^{q^j}$, while the corresponding entry of $\Lambda_M Df$ is zero. Thus, $m_k = 0$ for all $k \in (-j+P_f) \cup (1-j+P_f) \cup \cdots \cup (r-j+P_f) \cup (i-j+S_f)$. The remaining entries of $\pi[M^T Df + Df M]$ produce the relations $2m_{i-j} = 0$, $m_{i-j+1}^{q^j} + m_{i-j-1}^{q^{j+1}} = 0$, and so on the corresponding to the (i,k)th entry for $k \in P_f \cup (1+P_f) \cup \cdots \cup (r+P_f) \cup (i+S_f)$ is $m_{k-j}^{q^j}$. From these, one can derive that $m_k = 0$ for all $k \notin (i-j+[S_f \cup \{0\}])$.

By symmetry, you have that $m_k = 0$ for all $k \notin (r-s+[S_f \cup \{0\}])$ for all monomial summands $\alpha_{r,s} x^{q^r+q^s}$. Search for an element $g \in \mathbb{Z}_n$, where n is prime by assumption, such that g is in every such set. Since, for every $a \in S_f$ we have that $-a \in S_f$, a necessary condition is that S_f is closed under addition by g. Since every nonzero g is a generator of $\mathbb{Z}_n$, we must have that $g = 0$, otherwise it would contradict the fact that $D < q^{n/2}$. Thus, $Mx = m_0 x$, and after applying Theorem 2 from [63] in the case $m_0 \notin \mathbb{F}_q$ to conclude that $\pi \circ f$ is a quadratic monomial map. Since f is a nontrivial HFE polynomial, you have that $m_0 \in \mathbb{F}_q$. □

Dr. Smith-Tone and Mr. Daniels noted in [13], after they presented the above proof, that the conditions for the theorem are easily checked in a key generation algorithm. However, for smaller D, there may be difficulty in satisfying the conditions as well as a lack of entropy in the key space.

3.3 Cryptanalysis of HFEm

In [66], myself and Dr. Smith-Tone introduced a key recovery attack on HFEm. The basis for the attack is the Q-rank of the public key. This is in contrast to previous attacks on the scheme. One such attack in [4] focused on the Q-rank of the central map, which is part of the private key.

This attack depends on a key discovery, the ability to find an equivalent HFE instance of any given HFEm scheme. This technique was discovered by myself and Dr. Smith-Tone in [66]. This section walks the reader through our attack as well as providing a toy example for clarity. Unless specified otherwise, all theorems,definitions, and propositions were originally presented in [66].

3.3.1 Q-Rank

It is important to define a key quantity that is directly related to the security of "big field" schemes, the Q-rank. This quantity, when referring to the public key of a multivariate scheme, is defined as

Definition 3.5 (see [66]). *The Q-rank of any quadratic map $f(\overline{x})$ on $\mathbb{F}_q^n$ is the rank of the quadratic form $\phi^{-1} \circ f \circ \phi$ in $\mathbb{K}[X_0, \dots, X_{n-1}]$ via the identification $X_i = \phi(\overline{x})^{q^i}$.*

Quadratic form equivalence corresponds to matrix congruence, and thus the definition of the rank of a quadratic form is typically given as the minimum number of variables required to express an equivalent quadratic form. Since congruent matrices have the same rank, this quantity is equal to the rank of the matrix representation of this quadratic form, even in characteristic 2, where the quadratics x^{2q^i} are additive, but not linear for $q > 2$.

Q-rank is invariant under one-sided isomorphisms $f \mapsto f \circ U$, but is not invariant under isomorphisms of polynomials in general. The quantity that is often meant by the term Q-rank, but more properly called min-Q-rank, is the minimum

Q-rank among all nonzero linear images of f. This min-Q-rank is invariant under isomorphisms of polynomials and is the quantity relevant for cryptanalysis.

3.3.2 Previous Cryptanalysis of HFE

Since HFE was introduced in 1996, there have been three major techniques that have been developed that question its security. These techniques are as follows: the Kipnis-Shamir (KS) attack of [34], Faugére's direct algebraic attack using Gröbner bases in [25], and the minors modeling approach of the KS-attack of [4]. I will provide a short description of each attack for reference.

The attack created by Kipnis and Shamir in [34] is a key recovery attack. This method focuses on taking advantage of the fact that the central map f of an instance of HFE has a quadratic form, F over $\mathbb{K}$ as previously mentioned in section 3.3.1, that has low rank. While considering an odd characteristic case, you can view the homogeneous quadratic component of F in the following form:

$$\begin{bmatrix} x & x^q & \cdots & x^{q^{n-1}} \end{bmatrix} \begin{bmatrix} \alpha_{0,0} & \alpha'_{0,1} & \cdots & \alpha'_{0,d-1} & 0 & \cdots & 0 \\ \alpha'_{0,1} & \alpha_{1,1} & \cdots & \alpha'_{1,d-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \alpha'_{0,d-1} & \alpha'_{1,d-1} & \cdots & \alpha_{d-1,d-1} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x \\ x^q \\ \vdots \\ x^{q^{n-1}} \end{bmatrix},$$

where $\alpha'_{i,j} = \frac{1}{2}\alpha_{i,j}$ and $d = \lceil log_q(D) \rceil$. Next, the public key can be expressed as a quadratic polynomial G over a degree n extension using polynomial interpolation while keeping in mind that there is a linear map T^{-1} such that $T^{-1} \circ G$ has rank d, indicating that there exists a rank d matrix that is a $\mathbb{K}$-linear combination of the Frobenius powers of G. Hence, the question of recovering such a map T is an example of the MinRank problem over $\mathbb{K}$.

Note that the KS-attack was attempting to gain structural information to undermine the scheme. The attack proposed by Faugére in [25] does not follow this model. Instead of attempting to acquire structural information, Faugére's method attacks head on by using his F4 Gröbner basis algorithm. This attack was successful in breaking the HFE Challenge 1 proposed by Patarin in [48]. It is worth noting that F4 broke the challenge so easily due to the fact of the chosen parameters of HFE Challenge 1. Due to the scheme being defined over $GF(2)$ and using a degree 80 extension, HFE Challenge 1 was vulnerable. This is because the small base field greatly reduced the number of monomials of degree d, thus making a Gröbner basis attack very effective.

The previous two attacks were combined in the final cryptanalysis of HFE proposed in [4] which resulted in significant improvements. The technique demonstrated that a $\mathbb{K}$-linear combination of the *public* polynomials has low rank as a quadratic form over $\mathbb{K}$ due to a clever construction. Next, they set the unknown coefficients in $\mathbb{K}$ as variables, the polynomials representing $(d+1)\times(d+1)$ minors of such a linear combination, which must be zero due to the rank property, reside in $\mathbb{F}_q[t]$. This requires a Gröbner basis computation over $\mathbb{F}_q$ as well as the variety computed over $\mathbb{K}$. This technique is called minors modeling and dramatically improves the efficiency of the KS-attack. The complexity of the KS-attack with minors modeling is asymptotically $\mathcal{O}(n^{(\lceil log_q(D)\rceil+1)\omega})$, where $2\leq\omega\leq3$ is the linear algebra constant.

3.3.3 Key Recovery for HFEm

In this section I explain our key recovery attack on HFEm. The process is broken down into two main steps. The first is finding a related HFE instance of the

HFEm public key. This related instance will then be the focus. Then we discuss how to systematically solve for an equivalent private key for the orignal HFEm scheme.

Reduction of HFEm to HFE: Recall that by imposing the field equations we may always assume that any affine variety associated with HFE is contained in the finite field $\mathbb{K}$. Then we may use definition 3.1.

Recall that the public key of an HFEm scheme is constructed by truncating a full rank linear combination of the central polynomials. That is, with parenthetical emphasis, $P = \Pi(T \circ F \circ U)$. We now show that this singular linear transformation can be transported "past" the invertible transformation T and "absorbed" by the central map.

Lemma 3.1 (see [66])**.** *Let $\Pi \circ T$ be a corank a linear transformation on $\mathbb{F}_q^n$. There exist both a nonsingular linear transformation S and a degree q^a linear polynomial π such that $\Pi \circ T = S \circ \phi^{-1} \circ \pi \circ \phi$.*

Proof. Let V be the kernel of $\Pi \circ T$ and let $\pi = \mathcal{M}_V$. Note that $|V| = q^a$, thus $\mathcal{M}_V(x)$ has degree q^a and is of the form

$$x^{q^a} + c_{a-1}x^{q^{a-1}} + \cdots + c_1 x^q + c_0 x \text{ where } c_i \in \mathbb{K} \tag{3.9}$$

Now let $B_V = \{b_{n-a}, b_{n-a+1}, \ldots, b_{n-1}\}$ be a basis for V and extend this to a basis $B = \{b_0, \ldots, b_{n-1}\}$ of $\mathbb{F}_q^n$. Let M be the matrix transporting from the standard basis to B. Clearly the matrix representations of both $M^{-1}(\Pi \circ T)M$ and $M^{-1}(\phi^{-1} \circ \pi \circ \phi)M$ have the last a columns of 0.

Observe that there exist invertible matrices A and A', corresponding to row operations, such that both $AM^{-1}(\Pi \circ T)M$ and $A'M^{-1}(\phi^{-1} \circ \pi \circ \phi)M$ are in reduced echelon form; that is:

$$AM^{-1}(\Pi \circ T)M = \left[\begin{array}{c|c} I & 0 \\ \hline 0 & 0 \end{array}\right] = A'M^{-1}(\phi^{-1} \circ \pi \circ \phi)M \tag{3.10}$$

Solving for $\Pi \circ T$, we obtain

$$MA^{-1}A'M^{-1}(\phi^{-1} \circ \pi \circ \phi) = \Pi \circ T. \tag{3.11}$$

Let $S = MA^{-1}A'M^{-1}$ and the lemma is proven. □

Lemma 3.1 suggests the possibility of considering an HFEm public key as a full rank basis for the low rank image of a quadratic map. In fact, Lemma 3.1 is powerful enough to maintain a low degree bound for this map.

Theorem 3.4 (see [66]). *Let P be the public key of an HFE*$^-$$(q, n, D, a)$ *scheme. Then*

$$P' \coloneqq P \| \{p_{n-a}, p_{n-a+1} \dots, p_{n-1}\}$$

is a public key of an HFE$(q, n, q^a D)$ *scheme for any choice of* $p_i \in Span(P)$ *where* $i \in \{n-a, n-a+1, \dots, n-1\}$.

Proof. Let P be a public key for HFE$^-(q, n, D, a)$. Observe that P has the following form, $P = \Pi \circ T \circ F \circ U$ where $T, U : \mathbb{F}_q^n \to \mathbb{F}_q^n$ are affine transformations applied to an HFE(q, n, D) central map F. Let Π' be the natural embedding of Π as a linear map $\mathbb{F}_q^n \to \mathbb{F}_q^n$ obtained by composing the inclusion mapping $\mathbb{F}_q^{n-a} \to \mathbb{F}_q^n$. By Lemma 3.1, we can rewrite $P\|\{0, 0, \dots 0\}$ in the following way:

$$P\|\{0, 0, \dots 0\} = \Pi' \circ T \circ \phi^{-1} \circ f \circ \phi \circ U = S \circ \phi^{-1} \circ (\pi \circ f) \circ \phi \circ U, \tag{3.12}$$

where S is nonsingular and π is a linear polynomial of degree q^a.

Observe that $P\|\{0, 0, \dots 0\}$ now has the structure of an HFE$(q, n-a, q^a D)$, since the degree bound is increased by a factor of q^a; that is, $deg(\pi(f)) = deg(\pi)deg(f)$. Finally, construct $P' = P\|\{p_{n-a}, p_{n-a+1}, \dots, p_{n-1}\}$ where $p_i \in Span(P)$, possibly 0. Since the composition A of elementary row operations produces P' from $P\|\{0, 0 \dots, 0\}$, we obtain an HFE$(q, n, q^a D)$ key, $(AS, \pi \circ f, U)$. □

Theorem 3.4 indicates that HFEm, in some sense, *is* HFE with merely a slightly higher degree bound. Thus it is sensible to discuss recovering an equivalent key for an instance of HFEm as an HFE scheme. We can, in fact, do more and recover an equivalent HFEm key.

Key Recovery: Any HFE key recovery oracle $\mathcal{O}$, when given a public key P of an HFE instance recovers a private key of HFE "shape." By Theorem 3.4, such an oracle can recover a private key for the augmented public key P' which is also of HFE shape. We now show, however, that in this case, the key derived from $\mathcal{O}$ must preserve more structure.

Theorem 3.5 (see [66]). *Let P be a public key for an instance of HFE$^-(q,n,D,a)$ and let $P' = P\|\{p_{n-a},p_{n-a+1}\dots,p_{n-1}\}$ be a corresponding HFE(q,n,q^aD) public key. Further, let (T',f',U') be any private key of P'. Then the representation of f' as a quadratic form over $\mathbb{K}$ is block diagonal of the form:*

$$\mathbf{F}' = \begin{bmatrix} F_1' & 0 \\ 0 & 0 \end{bmatrix}, \tag{3.13}$$

where $F_1' = \left[f_{i,j}\right]_{i,j}$ is $(\lceil log_q(D)\rceil + a) \times (\lceil log_q(D)\rceil + a)$ and has the property that $f_{i,j} = 0$ if $|i-j| \geq \lceil log_q(D)\rceil$. That is, F_1' has only a diagonal "band" of nonzero values of width $2\lceil log_q(D)\rceil - 1$.

Proof. Let (T,f,U) be a private key for P as an instance of HFE$^-(q,n,D,a)$. By Theorem 3.4, one private key of P' has the form (T',f',U') where $f' = \pi\circ f$ and

$$\pi(x) = \sum_{i=0}^{a} b_i x^{q^i}.$$

Therefore,

$$\begin{aligned} f'(x) = \pi\circ f(x) &= \sum_{\substack{i\leq j \\ q^i+q^j\leq D}} \sum_{\ell=0}^{a} b_\ell \alpha_{i,j}^{q^\ell} x^{q^{i+\ell}+q^{j+\ell}} \\ &= \sum_{\substack{i,j\leq\lceil log_q(D)+a\rceil \\ |i-j|<\lceil log_q(D)\rceil}} f_{i,j} x^{q^i+q^j} \end{aligned}$$

Thus there exists one private key of the required form.

Denote by Frob_i the map raising all entries of a vector to the power q^i and let M_b be the linear map $x \mapsto bx$ for $b \in \mathbb{K}$. By the homogeneous case of [6, Theorem 4], for any second private key (T'', f'', U'') of P', we have for some integer $0 \le k < n$ and for some $a, b \in \mathbb{K}$ that

$$F'' = \text{Frob}_k \circ M_b \circ F' \circ M_a \circ \text{Frob}_{n-k}.$$

It is straightforward to check that the representation of F'' as a quadratic form has the shape of (3.13) with nonzero entries restricted to $|i - j| < \lceil log_q(D) \rceil$. □

Armed with Theorem 3.5, we are prepared to perform a full key recovery for an instance $P = \Pi \circ T \circ \phi^{-1} \circ f \circ \phi \circ U$ of HFEm. The strategy is simple. By way of Theorem 3.4, there exists an HFE instance with an equivalent public key. That is, there exists a $P' = T' \circ \phi^{-1} \circ f' \circ \phi \circ U'$ with T', U' invertible, f' of degree bounded by $q^a D$, and where the first $n - a$ public equations in P' form P while the remaining a equations are in the $\mathbb{F}_q$-linear span of P. We perform a key recovery on this instance of HFE via the best known attack, the KS-attack with minors modeling of [6]. Finally, we can recover a central map of degree bound D by way of the following theorem.

Theorem 3.6 (see [66]). *Let (T, f, U) be an HFE$^-(q, n, D, a)$ private key and let (T', f', U') be an equivalent HFE$(q, n, q^a D)$ key. Then a linear map T'' and a quadratic map f'' of degree bound D such that $\Pi \circ T'' \circ \phi^{-1} \circ f'' \circ \phi \circ U' = \Pi \circ T \circ \phi^{-1} \circ f \circ \phi \circ U$ can be recovered by solving two linear systems, the first of dimension a and the second of dimension $\binom{\lceil log_q(D) \rceil}{2}$.*

Proof. Let (T, f, U) be an HFE$^-(q, n, D, a)$ private key and let (T', f', U') be an equivalent HFE$(q, n, q^a D)$ key. Let $\mathbf{F}'$ denote the matrix representation of f' as a quadratic form over $\mathbb{K}$. Finally, let $d = \lceil log_q(D) \rceil$. By Theorem 3.5, $\mathbf{F}'$ has the diagonal band shape of width $2d-1$. From the proof of Theorem 3.4, there exists a linear map $\pi(x) = \sum_{i=0}^{a} p_i x^{q^i}$, where we may sacrifice monicity and insist $p_0 = 1$ for convenience, and a degree bound D quadratic function f'' such that the composition $\pi(f'') = f'$. Let $\mathbf{F}'' = (f''_{i,j})_{i,j}$ and $\widehat{\pi\mathbf{F}''}$ denote the matrix representations of f'' and $\pi \circ f''$, respectively, as quadratic forms over $\mathbb{K}$. Then we have $\mathbf{F}' = \widehat{\pi\mathbf{F}''}$. The (i,j)th entry of $\widehat{\pi\mathbf{F}''}$ is of the form

$$\sum_{\ell=0}^{a} p_\ell (f''_{i-\ell,j-\ell})^{q^\ell},$$

thus, since $\mathbf{F}'$ is known, we obtain a bilinear system of equations in the unknowns p_i and $f''_{i,j}$.

The insistence that $p_0 = 1$ allows us to recover the values of $f''_{0,j}$ without cost. We then note that due to the fact that $f''_{i,j} = 0$ when $max\{i,j\} \geq d$, the $(i, i+d-1)$th coefficients of $\widehat{\pi\mathbf{F}''}$ are $p_i(f''_{0,d-1})^{q^i}$ for $0 \leq i \leq a$. Thus, since $f''_{0,d-1}$ is known, we obtain *a linear* system of equations $f'_{i,i+d-1} = p_i(f''_{0,d-1})^{q^i}$ for $1 \leq i \leq a$ in the unknowns p_i, and can therefore solve for π. Once the values of p_i are known, the system of equations becomes linear in $f''_{i,j}$ for $i > 0$. Solving for the remaining unknown values can be done simply with the upper triangular segment from $(1,1)$ to $(d-1, d-1)$, of size $\binom{d}{2}$. □

To illustrate the attack in all of its steps, we have prepared a toy example provided in section 3.3.6.

3.3.4 Complexity of Attack

In this section we derive a tight complexity estimate of the key recovery attack for HFEm of Section 3.3.3. First, we expound upon the relationship between the computational complexity of of HFEm key recovery and that of HFE key recovery.

Theorem 3.7 (see [66]). *Let $\mathcal{O}$ be an HFE key recovery oracle that can recover a private key for any instance of HFE(q, n, D) in time $t(q, n, D)$. Then an equivalent HFE key for HFE$^-(q, n, D, a)$ can be recovered by $\mathcal{O}$ in time $t(q, n, q^a D)$.*

Proof. Let P be the public key for an instance of HFE$^-(q, n, D, a)$. Then make the following construction: $P' = P \| \{p_{n-a}, p_{n-a+1} \ldots, p_{n-1}\}$ where $p_i \in Span(P)$. By Theorem 3.4, P' is an instance of $HFE(q, n, q^a D)$. Thus $\mathcal{O}$ recovers an equivalent HFE key in time $t(q, n, q^a D)$. □

Thus, the complexity of deriving a key for the associated HFE scheme is bounded by the complexity of the best key recovery algorithm for HFE with a degree bound a factor of q^a larger. By Theorem 3.6, converting the recovered specially structured HFE$(q, n, q^a D)$ key into an equivalent HFE$^-(q, n, D, a)$ scheme is of complexity on the order of $\lceil log_q(D) \rceil^{2\omega}$. Since this quantity is very small, the key conversion is instantaneous for all practical parameters. Hence the complexity of the entire attack is bounded by $t(q, n, q^a D)$ from Theorem 3.7.

We can achieve a tight practical bound when specifying the oracle. Using the minors modeling approach to the KS-attack, which is the currently most successful algebraic attack on HFE, we can accurately determine the complexity of HFEm key recovery. Just as in HFE, the complexity of the attack is dominated by the MinRank calculation.

Proposition 3.4 (see [66]). *Let $d = \lceil log_q(D) \rceil$. The degree of regularity of the MinRank instance with parameters $(n, a + d, n - a)$ arising from minors modeling on the public key of HFE$^-(q, n, D, a)$ is the degree of the first negative term in the series*

$$H_r(t) = (1 - t)^{(n-a-d)^2 - n + a} \frac{det(\mathbf{A_{a+d}})}{t^{\binom{a+d}{2}}},$$

where $\mathbf{A_{a+d}}$ is the $(a + d) \times (a + d)$ matrix whose (i, j)-th entry is

$$a_{i,j} = \sum_{\ell=0}^{n-max\{i,j\}} \binom{n-i}{\ell} \binom{n-j}{\ell} t^{\ell}.$$

$\lceil log_q(D) \rceil$	2	3	4	5	6
d_{reg}	5	6	7	8	9

Table 3.1: The degree of regularity of the system arising from minors modeling on HFE$^-(q,n,D,a)$ with $a = 2$, $\lceil log_q(D) \rceil$ as indicated, and n sufficiently large.

Proposition 3.4 follows immediately from [24, Conjecture 3], which relies on the genericity conjecture [24, Conjecture 1] which is related to Fröberg's Conjecture, see [26]. With this proposition we can derive the degree of regularity for the MinRank instances for larger systems as well. Focusing on the case in which $a = 2$ we summarize the data in Table 3.1.

From these data we are prepared to make the following conjecture:

Conjecture 3.1 (see [66]). *The degree of regularity of the MinRank instance with parameters* $(n, a+d, n-a)$ *arising from minors modeling on the public key of HFE*$^-(q,n,D,a)$ *is*

$$d_{reg} = a + d + 1,$$

for all sufficiently large n.

Finally, under the above conjecture, we derive the complexity of our key recovery technique for HFEm.

Theorem 3.8 (see [66]). *The complexity of key recovery for HFE*$^-(q,n,D,a)$ *using the minors modeling variant of the KS-attack is*

$$\mathcal{O}\left(\binom{n-a+d_{reg}}{d_{reg}}^{\omega}\right) \sim \mathcal{O}\left(\binom{n+\lceil log_q(D)\rceil+1}{\lceil log_q(D)\rceil+a+1}^{\omega}\right).$$

3.3.5 Experiemental Results

We ran a series of experiments with Magma, see [7], on a 3.2 GHz Intel® Xeon™ CPU, testing the attack for a variety of values of q, n and D. In all cases, a valid private key was recovered. Table 3.2 summarizes some of our results for the asymptotically most costly step, the MinRank attack. The data support our complexity estimate of $\mathcal{O}\left(\binom{n+\lceil log_q(D)\rceil+1}{\lceil log_q(D)\rceil+a+1}^{\omega}\right)$.

a	$n = 8$	$n = 9$	$n = 10$	$n = 11$	$n = 12$
0	37	94	235	575	1269
1	166	535	1572	3653	3374
2	764	1254	6148	26260	97838

Table 3.2: Average time (in ms) for 100 instances of the MinRank attack on HFE$^-(3, n, 3^2 + 3^2 = 18, a)$ for various values of n and a.

3.3.6 Toy Example

To illustrate the attack, we present a complete key recovery for a small odd prime field instance of HFEm. We simplify the exposition by considering a homogeneous key.

Let $q = 7$, $n = 8$, $D = 14$ and $a = 2$. We construct the degree n extension $\mathbb{K} = \mathbb{F}_7[x]/\langle x^8 + 4x^3 + 6x^2 + 2x + 3\rangle$ and let $b \in \mathbb{K}$ be a fixed root of this irreducible polynomial.

We randomly select $f : \mathbb{K} \to \mathbb{K}$ of degree D,

$$f(x) = b^{4100689}x^{14} + b^{1093971}x^8 + b^{5273323}x^2,$$

and two invertible linear transformations T and U:

$$T = \begin{bmatrix} 2 & 1 & 0 & 3 & 5 & 0 & 3 & 2 \\ 6 & 2 & 1 & 3 & 4 & 2 & 5 & 1 \\ 0 & 2 & 5 & 1 & 3 & 1 & 4 & 3 \\ 3 & 2 & 6 & 4 & 5 & 3 & 4 & 4 \\ 6 & 4 & 2 & 1 & 0 & 5 & 0 & 0 \\ 0 & 3 & 3 & 6 & 5 & 1 & 1 & 3 \\ 0 & 3 & 0 & 4 & 3 & 6 & 1 & 5 \\ 4 & 3 & 2 & 6 & 1 & 1 & 6 & 3 \end{bmatrix}, \text{ and } U = \begin{bmatrix} 5 & 1 & 4 & 1 & 4 & 2 & 5 & 3 \\ 0 & 6 & 1 & 5 & 3 & 5 & 3 & 2 \\ 3 & 3 & 5 & 0 & 3 & 4 & 2 & 2 \\ 4 & 0 & 5 & 4 & 0 & 6 & 4 & 1 \\ 2 & 6 & 4 & 0 & 0 & 5 & 3 & 5 \\ 0 & 2 & 4 & 0 & 2 & 0 & 6 & 5 \\ 4 & 3 & 0 & 3 & 3 & 2 & 2 & 6 \\ 6 & 2 & 5 & 3 & 5 & 4 & 0 & 0 \end{bmatrix}.$$

Since $b^{1093971}/2 = b^{4937171}$, we have

$$\mathbf{F} = \begin{bmatrix} b^{5273323} & b^{4937171} & 0 & 0 & 0 & 0 & 0 & 0 \\ b^{4937171} & b^{4100689} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

We fix $\Pi : \mathbb{F}_q^8 \to \mathbb{F}_q^6$, the projection onto the first 6 coordinates. Then the public key $P = \Pi \circ T \circ F \circ U$ in matrix form over $\mathbb{F}_q$ is given by:

$$\mathbf{P_0} = \begin{bmatrix} 5 & 6 & 3 & 6 & 6 & 0 & 4 & 2 \\ 6 & 0 & 1 & 3 & 3 & 5 & 2 & 1 \\ 3 & 1 & 4 & 0 & 6 & 0 & 4 & 4 \\ 6 & 3 & 0 & 3 & 0 & 2 & 3 & 1 \\ 6 & 3 & 6 & 0 & 4 & 2 & 2 & 4 \\ 0 & 5 & 0 & 2 & 2 & 2 & 5 & 1 \\ 4 & 2 & 4 & 3 & 2 & 5 & 1 & 5 \\ 2 & 1 & 4 & 1 & 4 & 1 & 5 & 2 \end{bmatrix}, \mathbf{P_1} = \begin{bmatrix} 1 & 6 & 1 & 5 & 4 & 2 & 2 & 2 \\ 6 & 5 & 4 & 4 & 0 & 1 & 6 & 2 \\ 1 & 4 & 3 & 5 & 6 & 2 & 1 & 1 \\ 5 & 4 & 5 & 2 & 2 & 3 & 1 & 5 \\ 4 & 0 & 6 & 2 & 2 & 1 & 2 & 4 \\ 2 & 1 & 2 & 3 & 1 & 6 & 2 & 6 \\ 2 & 6 & 1 & 1 & 2 & 2 & 5 & 6 \\ 2 & 2 & 1 & 5 & 4 & 6 & 6 & 2 \end{bmatrix},$$

$$\mathbf{P_2} = \begin{bmatrix} 2 & 5 & 2 & 2 & 2 & 3 & 3 & 2 \\ 5 & 1 & 2 & 1 & 3 & 2 & 5 & 4 \\ 2 & 2 & 2 & 1 & 6 & 2 & 1 & 0 \\ 2 & 1 & 1 & 4 & 4 & 5 & 2 & 3 \\ 2 & 3 & 6 & 4 & 4 & 5 & 2 & 2 \\ 3 & 2 & 2 & 5 & 5 & 3 & 4 & 6 \\ 3 & 5 & 1 & 2 & 2 & 4 & 5 & 5 \\ 2 & 4 & 0 & 3 & 2 & 6 & 5 & 2 \end{bmatrix}, \mathbf{P_3} = \begin{bmatrix} 1 & 6 & 6 & 4 & 0 & 0 & 3 & 4 \\ 6 & 2 & 5 & 5 & 4 & 5 & 5 & 6 \\ 6 & 5 & 4 & 6 & 3 & 6 & 4 & 2 \\ 4 & 5 & 6 & 4 & 5 & 2 & 4 & 5 \\ 0 & 4 & 3 & 5 & 6 & 3 & 6 & 0 \\ 0 & 5 & 6 & 2 & 3 & 2 & 4 & 1 \\ 3 & 5 & 4 & 4 & 6 & 4 & 4 & 4 \\ 4 & 6 & 2 & 5 & 0 & 1 & 4 & 0 \end{bmatrix},$$

$$\mathbf{P_4} = \begin{bmatrix} 4 & 4 & 5 & 2 & 6 & 6 & 5 & 2 \\ 4 & 4 & 0 & 0 & 3 & 4 & 1 & 6 \\ 5 & 0 & 5 & 3 & 3 & 0 & 1 & 0 \\ 2 & 0 & 3 & 4 & 1 & 3 & 3 & 2 \\ 6 & 3 & 3 & 1 & 6 & 5 & 0 & 1 \\ 6 & 4 & 0 & 3 & 5 & 4 & 6 & 0 \\ 5 & 1 & 1 & 3 & 0 & 6 & 2 & 6 \\ 2 & 6 & 0 & 2 & 1 & 0 & 6 & 4 \end{bmatrix}, \mathbf{P_5} = \begin{bmatrix} 0 & 2 & 6 & 1 & 6 & 2 & 3 & 4 \\ 2 & 4 & 2 & 0 & 3 & 1 & 5 & 0 \\ 6 & 2 & 5 & 1 & 4 & 3 & 1 & 1 \\ 1 & 0 & 1 & 5 & 0 & 0 & 3 & 0 \\ 6 & 3 & 4 & 0 & 1 & 4 & 1 & 4 \\ 2 & 1 & 3 & 0 & 4 & 5 & 5 & 5 \\ 3 & 5 & 1 & 3 & 1 & 5 & 1 & 2 \\ 4 & 0 & 1 & 0 & 4 & 5 & 2 & 6 \end{bmatrix}$$

Recovering a Related HFE Key: This step in key recovery is a slight adaptation of the program of [6]. First, we recover the related private key of Theorem 3.5. To do this, we solve the MinRank instance on the above $6 = n - 2$ $n \times n$ matrices with target rank $\lceil log_q(D) \rceil + a = 2 + 2 = 4$. We may fix one variable to make the ideal generated by the 5×5 minors zero-dimensional. There are $n = 8$ solutions, each of which consists of the Frobenius powers of the coordinates of

$$v = (1, b^{5656746}, b^{3011516}, b^{3024303}, b^{1178564}, b^{1443785}).$$

The combination $L = \sum_{i=0}^{5} v_i \mathbf{P_i}$ is now a rank 4 matrix with entries in $\mathbb{K}$.

We next form $\widehat{v}$ from v by appending $a = 2$ random nonzero values from $\mathbb{K}$ to v. Now we compute

$$\phi^{-1} T'^{-1} \circ \phi = \sum_{i=0}^{8} \widehat{v}_i x^{q^i}.$$

Next we let K_i be the left kernel matrix of the $n - i$th Frobenius power of L for $i = 0, 1, \ldots, a + 1$. We then recover a vector w simultaneously in the right kernel of K_i for all i. For this example, each such element is a multiple in $\mathbb{K}$ of

$$w = (b^{4849804}, b^{3264357}, b^{4466027}, b^{638698}, b^{2449742}, b^{4337472}, b^{2752502}, b^{1186132}).$$

Then we may compute

$$\phi^{-1} \circ U \circ \phi = \sum_{i=0}^{8} w_i x^{q^i}.$$

At this point we can recover $\phi^{-1} \circ f' \circ \phi = T'^{-1} \circ P \circ U'^{-1}$, and have a full private key for the related instance HFE$(7, 8, 686)$. The transformations T' and U'

and the matrix representation of f' as a quadratic form over $\mathbb{K}$ are given by

$$T' = \begin{bmatrix} 1 & 4 & 4 & 5 & 4 & 5 & 5 & 2 \\ 0 & 6 & 6 & 0 & 4 & 4 & 5 & 5 \\ 0 & 5 & 0 & 4 & 2 & 0 & 0 & 3 \\ 0 & 4 & 4 & 2 & 5 & 6 & 6 & 6 \\ 0 & 3 & 6 & 2 & 5 & 6 & 0 & 0 \\ 0 & 2 & 0 & 4 & 4 & 6 & 2 & 2 \\ 0 & 1 & 5 & 5 & 0 & 5 & 2 & 6 \\ 0 & 3 & 3 & 3 & 6 & 5 & 2 & 2 \end{bmatrix}, U' = \begin{bmatrix} 6 & 2 & 1 & 4 & 4 & 4 & 1 & 6 \\ 1 & 6 & 0 & 2 & 3 & 0 & 4 & 2 \\ 2 & 5 & 3 & 6 & 3 & 3 & 0 & 4 \\ 0 & 5 & 6 & 5 & 4 & 1 & 4 & 2 \\ 6 & 5 & 3 & 5 & 4 & 6 & 3 & 2 \\ 0 & 4 & 6 & 1 & 4 & 0 & 1 & 5 \\ 6 & 0 & 2 & 3 & 6 & 5 & 6 & 3 \\ 5 & 2 & 0 & 4 & 1 & 2 & 4 & 5 \end{bmatrix}$$

$$\mathbf{F}' = \begin{bmatrix} b^{416522} & b^{5402526} & 0 & 0 & 0 & 0 & 0 & 0 \\ b^{5402426} & b^{3093518} & b^{5177024} & 0 & 0 & 0 & 0 & 0 \\ 0 & b^{5177024} & b^{5689467} & b^{5706144} & 0 & 0 & 0 & 0 \\ 0 & 0 & b^{5706144} & b^{3464750} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Recovery of Equivalent HFEm Key: Now we describe the full key recovery given the related HFE key. We know that there exists a degree $D = 14$ map

$f''(x) = f''_{0,0}x^2 + 2f''_{0,1}x^8 + f''_{1,1}x^{14}$ with associated quadratic form

$$\mathbf{F}'' = \begin{bmatrix} f''_{0,0} & f''_{0,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ f''_{0,1} & f''_{1,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

and a polynomial $\pi(x) = x + p_1x^7 + p_2x^{49}$ such that $f' = \pi \circ f''$. Thus we obtain the bilinear system of equations by equating $\mathbf{F}'$ to:

$$\widehat{\pi\mathbf{F}''} = \begin{bmatrix} f''_{0,0} & f''_{0,1} & 0 & 0 & 0 & 0 & 0 & 0 \\ f''_{0,1} & f''_{1,1} + p_1(f''_{0,0})^7 & p_1(f''_{0,1})^7 & 0 & 0 & 0 & 0 & 0 \\ 0 & p_1(f''_{0,1})^7 & p_1(f''_{1,1})^7 + p_2(f''_{0,0})^{49} & p_2(f''_{0,1})^{49} & 0 & 0 & 0 & 0 \\ 0 & 0 & p_2(f''_{0,1})^{49} & p_2(f''_{1,1})^{49} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

We clearly have the values of $f''_{0,0}$ and $f''_{0,1}$. Then the equations on the highest diagonal are linear in p_i. We obtain $\pi = x + b^{1948142}x^7 + b^{398370}x^{49}$ and continue to solve the now linear system to recover $f''(x) = b^{416522}x^2 + b^{1559326}x^8 + b^{1121420}x^{14}$.

We then obtain the matrix form of π over $\mathbb{F}_q$ and compose with T':

$$\widehat{\pi} = \begin{bmatrix} 2 & 6 & 6 & 0 & 2 & 2 & 5 & 5 \\ 6 & 3 & 5 & 3 & 1 & 4 & 5 & 0 \\ 5 & 2 & 6 & 0 & 6 & 6 & 6 & 1 \\ 1 & 1 & 3 & 6 & 4 & 1 & 1 & 6 \\ 5 & 6 & 2 & 4 & 6 & 6 & 1 & 6 \\ 5 & 3 & 1 & 5 & 0 & 1 & 0 & 4 \\ 3 & 2 & 1 & 3 & 3 & 1 & 3 & 5 \\ 4 & 2 & 1 & 1 & 1 & 4 & 4 & 2 \end{bmatrix}, T' \circ \widehat{\pi} = \begin{bmatrix} 0 & 0 & 1 & 2 & 0 & 5 & 4 & 0 \\ 1 & 2 & 4 & 4 & 2 & 1 & 0 & 4 \\ 0 & 2 & 2 & 1 & 1 & 6 & 1 & 0 \\ 3 & 3 & 1 & 0 & 6 & 3 & 2 & 0 \\ 0 & 1 & 3 & 1 & 0 & 2 & 2 & 2 \\ 3 & 4 & 5 & 0 & 1 & 3 & 4 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}.$$

Replacing the last two rows of $T' \circ \widehat{\pi}$ to make a full rank matrix produces T''. Then the original public key P is equal to $\Pi \circ T'' \circ \phi^{-1} \circ f'' \circ \phi \circ U'$.

CHAPTER 4
HFEV

In the previous chapter, the multivariate scheme of HFE and HFEm was discussed in detail. It was mentioned in section 3.1.2 that HFEv, the "vinegar" adaptation of HFE has received a great deal of attention. In this chapter, I go into detail on an analysis of the differential security of HFEv and HFEv$^-$. This analysis is a natural extension of the differential analysis of HFE and HFE$^-$ published in [13]. However, the situation is much more complex due to the nature of the vinegar adaptation. Expanding on those methods, I prove the following.

Theorem 4.1 (see [11]). *Let $\mathbb{K}$ be a degree n extension of the finite field $\mathbb{F}_q$. Let f be an $HFEv$ central map. With high probability, f has no nontrivial differential invariant structure.*

With a minimal augmentation of this method we extend this result to the case of $HFEv^-$.

Theorem 4.2 (see [11]). *Let f be an $HFEv$ central map and let π be a linear projection. With high probability, $\pi \circ f$ has no nontrivial differential invariant structure.*

Thus, with proper parameter selection, $HFEv^-$ is provably secure against differential adversaries. Together with the existant literature on resistance to algebraic and rank attacks, this security argument provides significant theoretical support for the security of aggressive $HFEv^-$ parameters, such as those presented in [53]. This chapter is organized by begining with details on the vinegar adaptation of HFE, then details on the symmetrical analysis, and ending with the invariant analysis.

4.1 HFEv and HFEv^- Scheme Description

For the vinegar modification, the construction follows the standard HFE process while going one step further: the addition of extra variables, $\tilde{x}_1, \ldots \tilde{x}_v$, which are to be assigned random values upon inversion. The effect of adding vinegar variables is that new quadratic terms, formed from both products of vinegar variables and HFE variables and products among vinegar variables, increase the rank of the public key. The central map of the $HFEv$ scheme has the form:

$$f(\mathbf{x}) = \sum_{\substack{i \leq j \\ q^i+q^j \leq D}} \alpha_{i,j} x^{q^i+q^j} + \sum_{\substack{i \\ q^i \leq D}} \beta_i(\tilde{x}_1, \ldots, \tilde{x}_v) x^{q^i} + \gamma(\tilde{x}_1, \ldots, \tilde{x}_v),$$

where $\alpha_{i,j} \in \mathbb{K}$, $\beta_i : \mathbb{F}_q^v \to \mathbb{K}$ is linear, and $\gamma : \mathbb{F}_q^v \to \mathbb{K}$ is quadratic.

In contrast to HFE, f is a vector-valued function mapping $\mathbb{F}_q^{n+v}$ to $\mathbb{F}_q^n$. The work of [34, 4, 13] show that representations of such functions over $\mathbb{K}$ are quite valuable. Thus it is beneficial to employ an augmentation of f, adding $n - v$ additional vinegar variables, and say $\hat{y} = \{\tilde{x}_1, \ldots, \tilde{x}_v, \ldots, \tilde{x}_n\}$, where $\tilde{x}_{v+1} = \tilde{x}_{v+2} = \ldots = \tilde{x}_n = 0$. Thus, our core map becomes

$$f(\mathbf{x}) = \hat{f}\left(\begin{array}{c} \hat{x} \\ \hat{y} \end{array} \right).$$

which algebraically identifies f as a bivariate function over $\mathbb{K}$. We may now write f in the following form:

$$f(x, y) = \sum_{\substack{0 \leq i \leq j < n \\ q^i+q^j \leq D}} \alpha_{ij} x^{q^i+q^j} + \sum_{\substack{0 \leq i,j < n \\ q^i \leq D}} \beta_{ij} x^{q^i} y^{q^j} + \sum_{0 \leq i \leq j < n} \gamma_{ij} y^{q^i+q^j}. \tag{4.1}$$

Here we see an obvious distinction among the types of monomials. We will label the monomials with α coefficients the "HFE monomials," those with β coefficients the "mixing monomials" and the monomials with γ coefficients the "vinegar monomials."

The $HFEv^{-}$ scheme uses the $HFEv$ primitive f above and augments the public key with the minus modifier. The minus modifier removes r of the public equations. This alteration is designed to destroy some of the information of the big field operations latent in the public key.

4.2 Linear Symmetry Analysis

4.2.1 HFEv

In our analysis, we will begin by considering the differential of our core map. From the perspective of our adversary, the discrete differential would be

$$D\hat{f}\left(\begin{bmatrix} \hat{a} \\ \hat{b} \end{bmatrix}, \begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}\right) = Df(a,b,x,y).$$

By the bilinearity of $D\hat{f}$ we see that Df is multi-affine; Df is affine in each of its inputs when the remaining inputs are fixed. Evaluating this differential we obtain

$$\begin{aligned} Df(a,b,x,y) = &\sum_{\substack{0\le i\le j<n \\ q^i+q^j\le D}} \alpha_{i,j}(x^{q^i}a^{q^j} + x^{q^j}a^{q^i}) && (4.2)\\ &+ \sum_{\substack{0\le i,j<n \\ q^i\le D}} \beta_{i,j}(x^{q^i}b^{q^j} + a^{q^i}y^{q^j}) && (4.3)\\ &+ \sum_{0\le i\le j<n} \gamma_{i,j}(y^{q^i}b^{q^j} + y^{q^j}b^{q^i}), && (4.4)\end{aligned}$$

noting that Df is a $\mathbb{K}$-bilinear form in $[a\ b]^T$ and $[x\ y]^T$. For ease of computation, we will choose the following representation for $\mathbb{K}$:

$$x \mapsto [x \ \ x^q \ \ x^{q^2} \ \ \dots \ \ x^{q^{n-1}}]^T.$$

Similarly, we may map our oil-vinegar vector as

$$[x \;\; y] \mapsto [x \;\; x^q \;\; x^{q^2} \;\; \dots \;\; x^{q^{n-1}} \;\; y \;\; y^q \;\; y^{q^2} \;\; \dots \;\; y^{q^{n-1}}]^T,$$

and Df is thus represented by the $2n \times 2n$ matrix where the (i,j)th and (j,i)th entries in the upper left $n \times n$ block are the coefficients $\alpha_{i,j}$, and the (i,j)th entries in the upper right block and the (j,i)th entries in the lower left block are the coefficients $\beta_{i,j}$, while the (i,j)th and the (j,i)th entries in the lower right block are the coefficients $\gamma_{i,j}$.

Note, that any $\mathbb{F}_q$-linear map $M : \mathbb{K} \to \mathbb{K}$ can be represented by $Mx = \sum_{i=0}^{n-1} m_i x^{q^i}$. Thus, as demonstrated in [13], under our representation,

$$M = \begin{pmatrix} m_0 & m_1 & \cdots & m_{n-1} \\ m_{n-1}^q & m_0^q & \cdots & m_{n-2}^q \\ \vdots & \vdots & \ddots & \vdots \\ m_1^{q^{n-1}} & m_2^{q^{n-1}} & \cdots & m_0^{q^{n-1}} \end{pmatrix}.$$

However, when viewing an $\mathbb{F}_q$-linear map over our vector $\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix}$, we may consider the $2n \times 2n$ matrix

$$\overline{M} = \begin{pmatrix} m_{00,0} & m_{00,1} & \cdots & m_{00,n-1} & m_{01,0} & m_{01,1} & \cdots & m_{01,n-1} \\ m_{00,n-1}^q & m_{00,0}^q & \cdots & m_{00,n-2}^q & m_{01,n-1}^q & m_{01,0}^q & \cdots & m_{01,n-2}^q \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ m_{00,1}^{q^{n-1}} & m_{00,2}^{q^{n-1}} & \cdots & m_{00,0}^{q^{n-1}} & m_{01,1}^{q^{n-1}} & m_{01,2}^{q^{n-1}} & \cdots & m_{01,0}^{q^{n-1}} \\ m_{10,0} & m_{10,1} & \cdots & m_{10,n-1} & m_{11,0} & m_{11,1} & \cdots & m_{11,n-1} \\ m_{10,n-1}^q & m_{10,0}^q & \cdots & m_{10,n-2}^q & m_{11,n-1}^q & m_{11,0}^q & \cdots & m_{11,n-2}^q \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots \\ m_{10,1}^{q^{n-1}} & m_{10,2}^{q^{n-1}} & \cdots & m_{10,0}^{q^{n-1}} & m_{11,1}^{q^{n-1}} & m_{11,2}^{q^{n-1}} & \cdots & m_{11,0}^{q^{n-1}} \end{pmatrix}.$$

For computational reference, we will label each row and column $modulo(n)$, i.e., each coordinate of the entry (i,j), will be represented by a residue class modulo n.

If we assume that f is vulnerable to a differential attack, then there exists a non-trivial linear mapping $\overline{M}$ such that the differential symmetry in (1) is satisfied. To compute such a symmetry inducing map requires the solution of $4n^2$ highly dependent but random equations in the $8n$ unknown coefficients of $\overline{M}$ and $\overline{\Lambda_M}$ over $\mathbb{K}$. Since trivial symmetries (such as multiplication by scalars) are exhibited by every map, we know that there exist nontrivial solutions. Even assuming unit time for $\mathbb{K}$-arithmetic operations, for realistic parameters this process is very inefficient; with the more realistic assumption of costly $\mathbb{K}$-arithmetic operations, this task is unsatisfactory in key generation.

To make the solution of such systems of equations more efficient, we derive the structure of the equations and develop a two step process for verifying trivial differential symmetric structure. The first step involves finding equations which only involve a subset of the variables. The existence of such equations is guaranteed by the degree bound of the HFE monomials. This information is then bootstrapped to eliminate many unknown coefficients of $\overline{M}$ resulting in a very small system of equations which can be solved explicitly.

We remark here that this methodology also suggests a method for estimating the probability of the existence of a differential symmetry for the $HFEv$ primitive. The existence of a nontrivial symmetry corresponds to systems for which the rank of the system of equations is less than $8n$. Under the heuristic that under row reduction these systems of equations behave like random $8n \times 8n$ matrices, we obtain a probability of roughly $1 - q^{-1}$ that the scheme has no nontrivial differential symmetry. We note that this heuristic is almost certainly false since trivial symmetries do exist. This quantity does represent a lower bound, however, and thus may offer support for larger base fields.

We begin by considering the entries of the matrix $\overline{M}^T Df + Df\overline{M}$. The contribution of any monomial $\alpha_{i,j}x^{q^i+q^j}$ to the ith row of $Df\overline{M}$ is given by

$$\begin{pmatrix}\alpha_{i,j}m^j_{00,-j} & \alpha_{i,j}m^j_{00,1-j} & \cdots & \alpha_{i,j}m^j_{00,-1-j} & \alpha_{i,j}m^j_{01,-j} & \alpha_{i,j}m^j_{01,1-j} & \cdots & \alpha_{i,j}m^j_{01,-1-j}\end{pmatrix}$$

while the contribution to the jth row is

$$\begin{pmatrix}\alpha_{i,j}m^i_{00,-i} & \alpha_{i,j}m^i_{00,1-i} & \cdots & \alpha_{i,j}m^i_{00,-1-i} & \alpha_{i,j}m^i_{01,-i} & \alpha_{i,j}m^i_{01,1-i} & \cdots & \alpha_{i,j}m^i_{01,-1-i}\end{pmatrix}.$$

By symmetry, the ith and and jth columns of $\overline{M}^T Df$ are the same as their respective rows.

It is clear that the rows and columns associated with coefficients of vinegar monomials as well as terms associated with mixing monomials may be represented similarly. However, it should be noted that those terms associated with mixing monomials will be multiplied by linear coefficients $m_{00,\cdot}$, $m_{01,\cdot}$, $m_{10,\cdot}$, and $m_{11,\cdot}$, while coefficients associated with vinegar variables are multiplied only by linear coefficients $m_{10,\cdot}$ and $m_{11,\cdot}$.

The above patterns can be extended to characterize the contribution to the ith row and jth row of monomials of the form $\beta_{i,j}x^{q^i}y^{q^j}$ and $\gamma_{i,j}y^{q^i+q^j}$, as well. We note, however, that γ coefficients interact with entries from the lower block matrices while β coefficients interact with coefficients from all block matrices.

Now that we have characterized the left side of (1), we will consider the entries of $\Lambda_{\overline{M}}Df$. For every monomial of f, say $\alpha_{i',j'}x^{q^i+q^j}$, $\beta_{r,s}x^{q^r}y^{q^s}$, or $\gamma_{u,v}y^{q^s+q^v}$, we have under the mapping of $\Lambda_{\overline{M}}$ terms of the form: $l_\ell\alpha^{q^\ell}_{i,j}x^{q^{i+\ell}+q^{j+\ell}}$, $l_\ell\beta^{q^{r+\ell}}_{r,s}x^{q^{s+\ell}}y^{q^j}$, and $l_\ell\gamma^{q^\ell}_{u,v}y^{q^{u+\ell}+q^{v+\ell}}$. Clearly, this results in every nonzero entry, say (r,s), of our Df matrix being raised to the power of q^ℓ and shifted along a forty-five degree angle to entry $(r+\ell, s+\ell)$. Thus, for each monomial in f there are two possible nonzero entries in the ith row, with possible overlap.

This discrete geometrical interpretation of the action of M and D on the coefficients of f is central to this analysis. A graphical representation of these relations is provided in Figure 4.1.

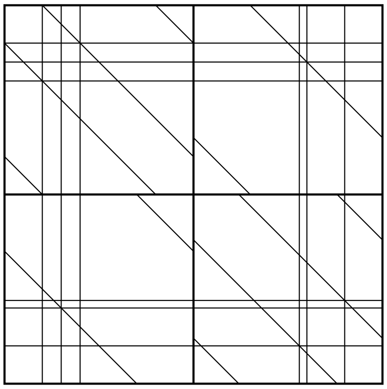


Figure 4.1: Graphical representation of the equation $M^T Df + DfM = \Lambda_M Df$ for the $HFEv$ (actually, vC^*) polynomial $f(x) = \alpha_{i,j} x^{q^i + q^j} + \beta_{r,s} x^{q^r} y^{q^s} + \gamma_{u,v} y^{q^u + q^v}$. Horizontal and vertical lines represent nonzero entries in $M^T Df + DfM$ while diagonal lines represent nonzero entries in $\Lambda_M Df$. We may consider this diagram as a genus 4 surface containing straight lines.

As in [13], the possibility of a differential symmetry can be determined by setting the matrix representation of $M^T Df + DfM$ equal to the matrix $\Lambda_M Df$. We will demonstrate an algorithm, given some specific constraints, that will help provide secure keys to be generated automatically.

Due to the structure of our M matrix, we need to work within each $m_{i,j}$ matrix independently. The following algorithm for $m_{0,0}$ extends very naturally to the other 3 matrices. For clarity, all m terms in description below are $m_{0,0}$ terms.

Let $\alpha_{i,j}, \beta_{r,s}, \gamma_{u,v}$ represent the coefficients of our monomials in our core map. Consider the ith row of $M^T Df + DfM$. For all w not occurring as a power of q of our HFE or mixing monomials in f, or difference of powers of q in an exponent of a monomial in f plus i, the (i, w) entry is $\alpha_{i,j} m_{w-j}^{q^j} = 0$ (resp. $\beta_{i,j} m_{w-j}^{q^j}$). Consider the rth row. For all w not occuring as an exponent of q in a vinegar monomial or as a difference of powers of q in an exponent of a monomial in f plus s, the (r,w)th entry is $\beta_{r,s}m_{k-s}^{q^s} = 0$. Hence, we can use those relations to look for non-zero entries of $m_{0,0}$.

After putting those relations into *HFEvKeyCheck* Algorithm, see Appendix C, you can generate a set for every i and r, exponents that occur in your core map. Each set provides a list of indices of all possible non-zero m's. For each index not occuring in any such set, the corresponding coefficient m must equal zero due to the fact that there must be a coordinate in the equation $M^TDf + DfM = \Lambda_M Df$ setting a constant multiple of m to zero. Thus, the intersection off all sets generated produces a list of all possible non-zero entries for the sub-matrix $m_{0,0}$.

Once this list is obtained, the variables shown to have value zero are eliminated from the system of equations. After repeating a similar algorithm for each of the remaining three submatrices a significantly diminished system of equations is produced which is then solved explicitly.

After running this algorithm with realistic values satisfying the above constraints and matching the parameter sizes of [53] along with using mild restrictions on the powers of the mixing and vinegar monomials, the only non-zero value obtained is m_0.

We note that it is possible that these restrictions, especially the restriction for these experiments on the number of monomials, place a lower bound on the number of vinegar variables required to achieve such a structure. On the other hand, with numerous small-scale experiments without parameter restrictions and using the full number of monomials we found that structurally the only nonzero value for the matrix $m_{0,0}$ is the m_0 term.

Since we have only a single non-zero term, our $m_{0,0}$ matrix is a diagonal matrix. A similar analysis for each of the remaining submatrices reveals the same structure. Thus we find that the only possible structure for $\overline{M}$ under these constraints satisfying a differential symmetry for $HFEv$ is

$$\overline{M} = \left[\begin{array}{c|c} cI & dI \\ \hline dI & cI \end{array}\right].$$

Furthermore, we can prove by way of Theorem 2 from [63], that the coefficients $c, d \in \mathbb{F}_q$.

We note that this map induces a trivial differential symmetry. To see this, note that the (nonpartial) differential of any bivariate function is bilinear in its vector inputs. Thus

$$\begin{aligned} Dg(\overline{M}[a\ b]^T, [x\ y]^T) &= Dg([ca + db\ da + cb]^T, [x\ y]^T) && (4.5) \\ &= Dg([ca + db\ cb + da]^T, [x\ y]^T) && (4.6) \\ &= Dg(c[a\ b]^T, [x\ y]^T) + Dg(d[b\ a]^T, [x\ y]^T) && (4.7) \\ &= cDg(a, b, x, y) + dDg(b, a, x, y) && (4.8) \\ &= (c + d)Dg(a, b, x, y). && (4.9) \end{aligned}$$

Consequently, for the parameters provided by *HFEvKeyCheck*, $HFEv$ provably has no nontrivial differential symmetric structure.

It should be noted that the restrictions provided on the powers of q of the monomials of our f does lower the entropy of our key space and likely raise the number of required vinegar variables to a level which is either unsafe or undesirable. However, there is still plenty of entropy with these restrictions and we obtain provable security against the differential symmetric attack. The restrictions provided are just a base line for this technique and our experiments with small scale examples indicate that even when we insist that every possible monomial satisfying the HFE degree bound is required to have a nonzero coefficient, the generalized algorithm still outputs only the trivial solution. Thus we can achieve provable security with minimal loss of entropy.

4.2.2 $HFEv^{-}$

Now, the algorithm extends naturally to $HFEv^{-}$. Every non-zero entry from the system generated by $HFEv$ is also in that generated by $HFEv^{-}$, but with a few more, see Figure 4.2. We choose a basis in which an example minus projection is a polynomial of degree q^2. For every ith row, we also have for any w not a power of $\alpha + n$ or $\beta + n$ where $n < 2$, the (i, w)th entry is $\alpha_{i,j} m_{w-j}^{q^j} = 0$. For the sth row, for all w not being a power of $\beta + n$ or $r + n$ where $n < 2$, the (s, w)th entry is $\beta_{r,s} m_{w-r}^{q^r} = 0$. A visualization is provided in Figure 4.2.

Again, we can use these relations, along with the relations described in the $HFEv$ system, to create a list of sets of all non-zero areas on $m_{0,0}$ using the algorithm *HFEv-KeyCheck*, see Appendix C. Each of these sets contains indices which are possibly non-zero, thus entries not in that set are definitively equal to zero.

By taking the intersection of all the sets, you can find the final locations of non-zero entries for our sub matrix $m_{0,0}$. In doing so, with realistic values from [53], the only non-zero value obtained is m_0. This again gives us security against symmetrical attacks by having M being a block matrix consisting of diagonal matrices with an argument similar to [13].

4.3 Invariant Analysis

4.3.1 Differential Invariants

Definition 4.1 (see [11]). *Let $f : \mathbb{F}_q^n \to \mathbb{F}_q^m$ be a function. A* differential invariant *of f is a subspace $V \subseteq \mathbb{K}$ with the property that there is a subspace $W \subseteq \mathbb{K}$ such that $dim(W) \leq dim(V)$ and $\forall A \in Span_{\mathbb{F}_q}(Df_i)$, $AV \subseteq W$.*

Informally speaking, a function has a differential invariant if the image of a

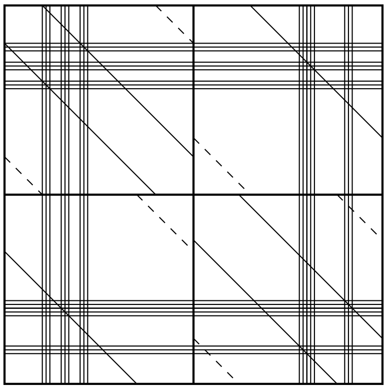


Figure 4.2: Graphical representation of the equation $M^TDf + DfM = \Lambda_M Df$ for the $HFEv^-$ with the minus modifier given by the projection $\pi(x) = x^{q^2} + \rho x^q + \tau x$. Horizontal and vertical lines represent nonzero entries in $M^TDf + DfM$ while diagonal lines represent nonzero entries in $\Lambda_M Df$. We note that each triple of lines corresponds to a single monomial in the central map.

subspace under all differential coordinate forms lies in a fixed subspace of dimension no larger. This definition captures the notion of *simultaneous invariants*, subspaces which are simultaneously invariant subspaces of Df_i for all i, and detects when large subspaces are acted upon linearly.

If we assume the existence of a differential invariant V, we can define a corresponding subspace $V^\perp$ as the set of all elements $x \in \mathbb{K}$ such that the dot product $\langle x, Av\rangle = 0 \ \forall v \in V, \forall A \in Span(Df_i)$. We note that this is not the standard definition of an orthogonal complement. $V^\perp$ is not the set of everything orthogonal to V, but rather everything orthogonal to AV, which may or may not be in V. By definition, it is clear that V and $V^\perp$ satisfy the relation

$$dim(V) + dim(V^\perp) \geq n.$$

Assume there is a differential invariant $V \subseteq \mathbb{F}_q^n$, and choose linear maps $M : \mathbb{F}_q^n \to V$ and $M^\perp : \mathbb{F}_q^n \to V^\perp$. For any differential-coordinate-form, we have

$$[Df(M^\perp y, Mx)]_i = (M^\perp y)^T \left(Df_i(Mx)\right) \tag{4.10}$$

Since $M^{\perp}y$ is in $V^{\perp}$, and $Df_iMx \in AV$, we must then have that

$$[Df(M^{\perp}y, Mx)]_i = (M^{\perp}a)^T \left(Df_i(Mx)\right) = 0 \tag{4.11}$$

Thus, as derived in [51],

$$\forall y, x \in \mathbb{F}_q^n, Df(M^{\perp}y, Mx) = 0 \quad \text{or equivalently,} \quad Df(M^{\perp}\mathbb{F}_q^n, M\mathbb{F}_q^n) = 0 \tag{4.12}$$

This relation restricts the structure of M and $M^{\perp}$, and provides a direct means of classifying the differential invariant structure of f.

We follow an analogous strategy to that of [13], adapted to the structure of the central $HFEv^{-}$ map f. First, we recall proposition 3.1. Without loss of generality we assume that $rank(M^{\perp}) \leq rank(M)$. If the ranks are equal, then we may apply the proposition and write $M^{\perp} = SMT$, with S and T nonsingular. If $rank(M^{\perp}) < rank(M)$, compose M with a singular matrix X so that $rank(XM) = rank(M^{\perp})$, and then apply the above result so that $M^{\perp} = S(XM)T$. Then we can express $M^{\perp} = S'MT$, where S' is singular. Restating our differential result (4.12) in this manner, we have that if $M^{\perp} = SMT$, and $M : \mathbb{F}_q^{n+v} \to V$, then

$$\forall x, y \in \mathbb{F}_q^n, Df(SMTy, MTx) = 0. \tag{4.13}$$

4.3.2 Minimal Generators over Intermediate Subfield

For lack of a good reference, we prove the following statement about the structure of the coordinate ring of a subspace of an extension field over an intermediate extension.

Lemma 4.1 (see [11]). *Let $\mathbb{L}/\mathbb{K}/\mathbb{F}_q$ be a tower of finite extensions with $|\mathbb{L} : \mathbb{K}| = m$ and $|\mathbb{K} : \mathbb{F}_q| = n$. Let V be an $\mathbb{F}_q$-subspace of $\mathbb{L}$. Then $I(V)$ has m multivariate generators over $\mathbb{K}$ of the form*

$$\mathcal{M}_V^{(k)}(x_0, \ldots, x_{m-1}) = \sum_{\substack{0 \leq i < n \\ 0 \leq j < m}} a_{ijk} x_j^{q^i}.$$

Proof. Choose a basis $\{\overline{e_0} = \overline{1}, \overline{e_1}, \ldots, \overline{e_{m-1}}\}$ for $\mathbb{L}$ over $\mathbb{K}$. Since V is an $\mathbb{F}_q$-subspace of $\mathbb{L}$, the minimal polynomial of V over $\mathbb{L}$, $\mathcal{M}_V(\overline{X}) = \sum_{i=0}^{mn-1} \overline{\alpha_i}\overline{X}^{q^i}$, is $\mathbb{F}_q$-linear. Note that the operations of addition and left multiplication by elements in $\mathbb{L}$ are $\mathbb{K}$-linear, whereas the Frobenius maps are merely $\mathbb{F}$-linear.

Now, since $\mathcal{M}_V(\overline{X})$ is linear it is additive, hence

$$\mathcal{M}_V(\overline{X}) = \mathcal{M}_V\left(\begin{bmatrix} x_0 \\ \vdots \\ x_{m-1} \end{bmatrix}\right) = \sum_{i=0}^{m-1} \mathcal{M}_V(x_i\overline{e_i}).$$

In each summand of $\mathcal{M}_V(x_j\overline{e_j})$, we have

$$(x_j\overline{e_j})^{q^i} = x_j^{q^i}\overline{e_j}^{q^i} = x_j^{q^i}\sum_{i=0}^{m-1} r_i\overline{e_i}$$

for some $r_0, \ldots, r_{m-1} \in \mathbb{K}$. As a vector over $\mathbb{K}$ this quantity is

$$\begin{bmatrix} r_0 x_j^{q^i} \\ \vdots \\ r_{m-1} x_j^{q^i} \end{bmatrix}.$$

Thus $\mathcal{M}_V(x_j\overline{e_j})$ is an m-dimensional vector of $\mathbb{K}$-linear combinations of $x_j, x_j^q, \ldots, x_j^{q^{n-1}}$. Thus $\mathcal{M}_V(\overline{X})$ is of the form

$$\mathcal{M}_V(\overline{X}) = \begin{bmatrix} \mathcal{M}_V^{(0)}(x_0, \ldots, x_{m-1}) \\ \vdots \\ \mathcal{M}_V^{(m-1)}(_0, \ldots, x_{m-1}) \end{bmatrix} = \begin{bmatrix} \sum_{\substack{0\le i<n \\ 0\le j<m}} a_{ij0}x_j^{q^i} \\ \vdots \\ \sum_{\substack{0\le i<n \\ 0\le j<m}} a_{ij(m-1)}x_j^{q^i} \end{bmatrix},$$

as required. □

We note that the minimal polynomials studied in [13] correspond to the special case of the above lemma in which $m = 1$. Given our characterization from Section 4.1 of the central map of $HFEv^-$ as a bivariate polynomial over $\mathbb{K}$, we are primarily interested in the $m = 2$ case of Lemma 4.1.

4.3.3 HFEv

As in [13], we consider $Df(SMTa, MTx)$, where T is nonsingular, S is a possibly singular map which sends V into $V^{\perp}$ and $M : k \to k$ is a projection onto V. Without loss of generality we'll assume that M projects onto V. Then MT is another projection onto V. SMT is a projection onto $V^{\perp}$. An important distinction is that for this case, the a and x above are actually two dimensional vectors over k. Thus $dim(V) + dim(V^{\perp}) \geq n$.

of Theorem 4.1. Let us denote by $[\hat{x}\ \hat{y}]^T$ the quantity $MT[x\ y]^T$.

Suppose we have

$$f(x,y) = \sum_{\substack{0 \leq i \leq j < n \\ q^i + q^j \leq D}} \alpha_{ij} x^{q^i + q^j} + \sum_{\substack{0 \leq i,j < n \\ q^i \leq D}} \beta_{ij} x^{q^i} y^{q^j} + \sum_{0 \leq i \leq j < n} \gamma_{ij} y^{q^i + q^j}.$$

Applying the differential (w.r.t. the vector $[x\ y]^T$) as described in Section 4.2, we obtain:

$$\begin{aligned} Df(a,b,x,y) = & \sum_{\substack{0 \leq i \leq j < n \\ q^i + q^j \leq D}} \alpha_{ij} \left(a^{q^i} x^{q^j} + a^{q^j} x^{q^i} \right) && (4.14) \\ & + \sum_{\substack{0 \leq i,j < n \\ q^i \leq D}} \beta_{ij} \left(a^{q^i} y^{q^j} + x^{q^i} b^{q^j} \right) && (4.15) \\ & + \sum_{0 \leq i \leq j < n} \gamma_{ij} \left(b^{q^i} y^{q^j} + b^{q^j} y^{q^i} \right). && (4.16) \end{aligned}$$

Substituting $SMT[a\ b]^T$ and $MT[x\ y]^T$, we derive

$$Df(S[\hat{a}\ \hat{b}]^T, \hat{x}, \hat{y}) = Df(S_{11}\hat{a} + S_{12}\hat{b}, S_{21}\hat{a} + S_{22}\hat{b}, \hat{x}, \hat{y}).$$

For notational convenience let $\hat{\hat{a}}$= $S_{11}\hat{a} + S_{12}\hat{b}$ and $\hat{\hat{b}}$= $S_{21}\hat{a} + S_{22}\hat{b}$. Plugging in these values in the previous equation we get

$$
\begin{aligned}
Df(\hat{\hat{a}}, \hat{\hat{b}}, \hat{x}, \hat{y}) = \sum_{\substack{0 \leq i \leq j < n \\ q^i + q^j \leq D}} & \alpha_{ij} \left((\hat{\hat{a}})^{q^i} \hat{x}^{q^j} + (\hat{\hat{a}})^{q^j} \hat{x}^{q^i} \right) && (4.17) \\
+ \sum_{\substack{0 \leq i, j < n \\ q^i \leq D}} & \beta_{ij} \left((\hat{\hat{a}})^{q^i} \hat{y}^{q^j} + \hat{x}^{q^i} (\hat{\hat{b}})^{q^j} \right) && (4.18) \\
+ \sum_{0 \leq i \leq j < n} & \gamma_{ij} \left((\hat{\hat{b}})^{q^i} \hat{y}^{q^j} + (\hat{\hat{b}})^{q^j} \hat{y}^{q^i} \right). && (4.19)
\end{aligned}
$$

In contrast to the situation with HFE, these monomials are not necessarily independent. By Lemma 4.1, the generators of $I(V)$ have the form

$$
\sum_{0 \leq i < n} r_{ij} x^{q^i} + \sum_{0 \leq i < n} s_{ij} y^{q^i} \text{ for } j \in \{1, 2\},
$$

where $r_{ij}, s_{ij} \in \mathbb{K}$. Clearly, these expressions evaluate to zero on $(\hat{x}, \hat{y})$. Evaluating (4.17) modulo $I(V)$ (only on the variables $\hat{x}$ and $\hat{y}$), we obtain:

$$
\begin{aligned}
Df(\hat{\hat{a}}, \hat{\hat{b}}, \hat{x}, \hat{y}) = \sum_{\substack{0 \leq i < n \\ 0 \leq j < d_x}} & \left[\alpha'_{ij} (\hat{\hat{a}})^{q^i} + \beta'_{ij} (\hat{\hat{b}})^{q^i} \right] \hat{x}^{q^j} && (4.20) \\
+ \sum_{\substack{0 \leq i < n \\ 0 \leq j < d_y}} & \left[\gamma'_{ij} (\hat{\hat{a}})^{q^i} + \delta'_{ij} (\hat{\hat{b}})^{q^i} \right] \hat{y}^{q^j}, && (4.21)
\end{aligned}
$$

where d_x and d_y are the largest powers of $\hat{x}$ (resp. $\hat{y}$) occuring. After the reduction modulo $I(V)$, the remaining monomials $\hat{x}, \ldots, \hat{x}^{q^{d_x}}$ and $\hat{y}, \ldots, \hat{y}^{q^{d_y}}$ are independent. Thus, for $Df(\hat{\hat{a}}, \hat{\hat{b}}, \hat{x}, \hat{y}) = 0$, each polynomial expression multiplied by a single $\hat{x}^{q^j}$ or $\hat{y}^{q^j}$ must be identically zero, that is to say that for all $0 \leq j \leq d_x$

$$
\sum_{0 \leq i < n} \left[\alpha'_{ij} (\hat{\hat{a}})^{q^i} + \beta'_{ij} (\hat{\hat{b}})^{q^i} \right] = 0 \tag{4.22}
$$

and for all $0 \leq j \leq d_y$

$$
\sum_{0 \leq i < n} \left[\gamma'_{ij} (\hat{\hat{a}})^{q^i} + \delta'_{ij} (\hat{\hat{b}})^{q^i} \right] = 0. \tag{4.23}
$$

The left hand sides of (4.22) and (4.23) are $\mathbb{F}$-linear functions in $S[\hat{a} \ \hat{b}]^T$. Thus we can express each such equality over $\mathbb{F}$ as

$$
LS \left[\hat{a}_0 \cdots \hat{a}_{n-1} \ \hat{b}_0 \cdots \hat{b}_{n-1} \right]^T = 0,
$$

where L is an $n \times 2n$ matrix with entries in $\mathbb{F}$. We note specifically that the coefficients of L depend on V and the choices of coefficients in the central map f. For randomly chosen coefficients retaining the $HFEv$ structure, we expect an L derived from an equation of the form (4.22) or (4.23) to have high rank with very high probability, more than $1 - q^{-n}$. Thus the dimension of the intersections of the nullspaces of each L is zero with probability at least $1 - 2q^{-n}$.

Clearly, the condition for these equations to be satisfied is that S sends V to the intersection of the nullspaces of each such L. Thus S is with high probability the zero map on V and so $V^{\perp} = \{0\}$. This generates a contradiction, however, since $2n \leq dim(V) + \dim(V^{\perp}) < 2n$. Thus, with probability greater than $1 - 2q^{-n}$, f has no nontrivial differential invariant structure. □

4.3.4 $HFEv^{-}$

The situation for $HFEv^{-}$ is quite similar, but the probabilities are slightly different. Specifically one must note that since the condition of being a differential invariant is a condition on the span of the public differential forms, under projection this condition is weaker and easier to satisfy. For specificity, we consider the removal of a single public equation, though, critically, a very similar though notationally messy analysis is easy to derive in the general case.

We may model the removal of a single equation as a projection of the form $\pi(x) = x^q + x$ applied after the central map.

Proof of Theorem 4.2. Consider

$$\pi(f(x,y)) = \sum_{\substack{0\leq i\leq j<n \\ q^i+q^j\leq D}} \alpha_{ij}x^{q^i+q^j} + \sum_{\substack{0\leq i,j<n \\ q^i\leq D}} \beta_{ij}x^{q^i}y^{q^j} + \sum_{0\leq i\leq j<n} \gamma_{ij}y^{q^i+q^j} \tag{4.24}$$

$$+ \sum_{\substack{0\leq i\leq j<n \\ q^i+q^j\leq D}} \alpha_{ij}^q x^{q^{i+1}+q^{j+1}} + \sum_{\substack{0\leq i,j<n \\ q^i\leq D}} \beta_{ij}^q x^{q^{i+1}}y^{q^{j+1}} + \sum_{0\leq i\leq j<n} \gamma_{ij}^q y^{q^{i+1}+q^{j+1}}. \tag{4.25}$$

Taking the differential, we obtain

$$
\begin{align}
D(\pi \circ f)(\hat{a}, \hat{\hat{b}}, \hat{x}, \hat{y}) = \sum_{\substack{0 \le i \le j < n \\ q^i + q^j \le D}} & \alpha_{ij} \left((\hat{a})^{q^i} \hat{x}^{q^j} + (\hat{a})^{q^j} \hat{x}^{q^i} \right) \tag{4.26} \\
+ \sum_{\substack{0 \le i, j < n \\ q^i \le D}} & \beta_{ij} \left((\hat{a})^{q^i} \hat{y}^{q^j} + \hat{x}^{q^i} (\hat{\hat{b}})^{q^j} \right) \tag{4.27} \\
+ \sum_{0 \le i \le j < n} & \gamma_{ij} \left((\hat{\hat{b}})^{q^i} \hat{y}^{q^j} + (\hat{\hat{b}})^{q^j} \hat{y}^{q^i} \right) \tag{4.28} \\
+ \sum_{\substack{0 \le i \le j < n \\ q^i + q^j \le D}} & \alpha_{ij}^{q} \left((\hat{a})^{q^{i+1}} \hat{x}^{q^{j+1}} + (\hat{a})^{q^{j+1}} \hat{x}^{q^{i+1}} \right) \tag{4.29} \\
+ \sum_{\substack{0 \le i, j < n \\ q^i \le D}} & \beta_{ij}^{q} \left((\hat{a})^{q^{i+1}} \hat{y}^{q^{j+1}} + \hat{x}^{q^{i+1}} (\hat{\hat{b}})^{q^{j+1}} \right) \tag{4.30} \\
+ \sum_{0 \le i \le j < n} & \gamma_{ij}^{q} \left((\hat{\hat{b}})^{q^{i+1}} \hat{y}^{q^{j+1}} + (\hat{\hat{b}})^{q^{j+1}} \hat{y}^{q^{i+1}} \right). \tag{4.31}
\end{align}
$$

Again, we may evaluate modulo $I(V)$ and collect the terms for the distinct powers of $\hat{x}$ and $\hat{y}$. By the independence of these monomials we obtain the relations

$$
\begin{align}
\sum_{0 \le i < n} \left[\alpha_{ij}''(\hat{a})^{q^i} + \beta_{ij}'(\hat{\hat{b}})^{q^i} \right] = 0 \tag{4.32} \\
\sum_{0 \le i < n} \left[\gamma_{ij}''(\hat{a})^{q^i} + \delta_{ij}'(\hat{\hat{b}})^{q^i} \right] = 0. \tag{4.33}
\end{align}
$$

At this point, the analysis proceeds exactly as in the case of $HFEv$. We once again arrive at the conclusion that with high probability S is the zero map on V, contradicting the existence of a differential invariant. We note here that this analysis works for any projection, though the exact values of the α_{ij}'' and γ_{ij}'' depend on the specific projection and the structure of f. □

4.4 Closing Remarks for HFEv and $HFEv^-$

$HFEv^-$ is rapidly approaching twenty years of age and stands as one of the oldest post-quantum signature schemes remaining secure. With the new parameters suggested in [53], $HFEv^-$ has metamorphosed from the very slow form of QUARTZ into a perfectly reasonable option for practical and secure quantum-resistant signatures.

Our analysis contributes to the confidence and optimism which $HFEv^-$ inspires. By elucidating the differential structure of the central map of $HFEv^-$, we have verified that a class of attacks which has proven very powerful against multivariate schemes in the past cannot be employed against $HFEv^-$. In conjunction with the careful analysis of the degree of regularity and Q-rank of the scheme already present in the literature, we have succeeded in showing that $HFEv^-$ is secure against every type of attack known. If the future holds a successful attack against $HFEv^-$ it must be by way of a fundamentally new advance.

CHAPTER 5
HFERP

5.1 Introduction

5.1.1 Recent History

While there may be many trustworthy candidates for multivariate signatures, such as UOV [32], Rainbow [16], and Gui [54], developing multivariate schemes for encryption has been a bit of a struggle. While some older ideas have have been reborn with better parameter sets due to the advancement of the science, such as applying HFE-, see [48], to encryption, most of the surviving multivariate encryption schemes are relatively young.

In the last few years, there have been a few new proposals for multivariate encryption, mostly following the idea that it is easier to hide the structure of an injective mapping into a large codomain than to hide the structure of a bijection, as is needed for any encryption mapping into a codomain of the same size as the domain. The ABC Simple Matrix encryption scheme of [64, 19] and ZHFE, see [55] are examples of this idea. Most of these encryption ideas, both new and old, have inspired recent surprising cryptanalyses that affect parameter selection or outright break the scheme, see [39, 42, 41, 10, 67], for example.

Such a tale describes the life of SRP, see [70], the design of which aimed to be very efficient and holds a comparably small blow up factor between the plaintext and ciphertext sizes. The scheme also claimed security against attacks efficient against the Square and Rainbow schemes by combining them into one. Unfortunately, SRP is also the victim of a new cryptanalysis, see [52]. The attack exploits the low Q-rank of the Square map, a vulnerability inherited by the public key. A modified MinRank attack was able to pull apart the Square polynomials from the Rainbow and Plus polynomials in the public key.

5.1.2 Our Contribution

We present a new composite scheme in the manner of SRP by replacing the weaker Square layer with an HFE polynomial of higher Q-rank and finding the correct balance in the sizes of the HFE, Rainbow and Plus layers for efficiency and security. We call our scheme HFERP. We further establish the complexity of the relevant attack models: the algebraic attack, the MinRank attack, and the invariant attack.

5.1.3 Organization

The remainder of this chapter is organized as follows. In the next section, we present isomorphisms of polynomials and describe the structure of HFE and SRP. The subsequent section reviews the Q-rank of ideals in polynomial rings and discusses invariant properties of Q-rank and min-Q-rank. In section 5.3, we review more carefully the previous cryptanalyses of HFE and SRP. We then present HFERP in the next section. Section 5.5 discusses the complexity of all known relevant attacks on HFERP. Our choice of parameters to optimize security and performance along with experimental results are then presented in the following section. Finally, we conclude discussing why a similar approach to SRP seems to produce such a different technology in HFERP.

5.2 Component Descriptions

5.2.1 Rainbow

The Rainbow scheme is a generalization of Patarin's UOV, see [32]. The key idea, introduced by Ding, see [16], was constructing multiple layers of UOV.

Let $\mathbb{F}$ be a finite field with a degree n extension $\mathbb{F}^n$. Let $\mathcal{V} = \{1, 2, \ldots, n\}$. For a chosen u, let $v_1, \ldots, v_u$ be integers such that $0 < v_1 < \cdots < v_u = n$ and let $\mathcal{V}_l = \{1, \ldots, v_l\}$ for each $l \in \{1, \ldots, u\}$. Note that $|\mathcal{V}_i| = v_i$.

Let $o_i = v_{i+1} - v_i$ for each $i \in \{1, \ldots, u-1\}$ and $\mathcal{O}_i = S_{i+1} - S_i$ for each $i \in \{1, \ldots, u-1\}$. Define P_l to be the space generated by the span of polynomials of the following form:

$$f(x_1, \ldots, x_n) = \sum_{i \in \mathcal{O}_l, j \in \mathcal{V}_l} \alpha_{i,j} x_i x_j + \sum_{i,j \in \mathcal{V}_l} \beta_{i,j} x_i x_j + \sum_{i \in \mathcal{V}_l} \gamma_i x_i + \eta$$

One can refer to the previous constructions using the following terminology: $\mathcal{O}$ is the collection of oil variables, $\mathcal{V}$ is the collection of vinegar variables, and a polynomial $f \in P_l$ is an l-th layer Oil and Vinegar polynomial.

The Rainbow map $F : \mathbb{F}^n \to \mathbb{F}^{n-v_1}$ is defined as (with $x_1, \ldots, x_n$ being referred to as $\bar{x}$ for convenience)

$$F(\bar{x}) = (\tilde{F}_1(\bar{x}), \ldots, \tilde{F}_{u-1}(\bar{x})) = (F_1(\bar{(}x), \ldots, F_{n-v_1}(\bar{x})$$

where each $\tilde{F}_i$ consists of o_i randomly chosen quadratic polynomials from P_i. F is a Rainbow polynomial map with $u-1$ layers. The public key is generated in the usual fashion by applying two affine transformations, T and U, where $T : \mathbb{F}^{n-v_1} \to \mathbb{F}^{n-v_1}$ and $U : \mathbb{F}^n \to \mathbb{F}^n$: $T \circ F \circ U$

5.2.2 SRP

In Section 5.4, we present in detail the construction of our proposed scheme, HFERP. For reference, we will include the Square Map definition as well as method of inversion presented in the original SRP paper, see [70].

Instead of using the HFE core map described in section 5.4, SRP uses the Squaring map where the Square component is defined as $\mathcal{F}_S : \mathbb{F}_q^{n'} \to \mathbb{F}_q^d$ (where $q^d + 1$ is divisible by 4) and it is the result of the following composition:

$$\mathbb{F}_q^{n'} \xrightarrow{\pi_d} \mathbb{F}_q^d \xrightarrow{\phi} \mathbb{K} \xrightarrow{X \mapsto X^2} \mathbb{K} \xrightarrow{\phi^{-1}} \mathbb{F}_q^d$$

Upon inversion step 3, the user would compute

$$R_{1,2} = \pm X^{(q^d+1)/4}$$

and use it to find $\mathbf{y} = (y_1^{(i)}, \dots, y_d^{(i)}) = \phi^{-1}(R_i) \in \mathbb{F}_q^d$. The choice of the Square map was made because of the speed of inversion it provided when compared to any other quadratic maps. Unfortunately, due to this choice, SRP was quickly broken in [52] by isolating the squaring public polynomials and exploiting its low Q-rank.

5.3 Previous Cryptanalysis of Relevant Schemes

SRP was a designed as a concatenation of two known multivariate schemes and a scheme modifier. The first component was Square, see [12], which can be seen as a degenerate version of HFE. The second component was oil-and-vinegar (OV) or, more generally, Rainbow, see [46, 16]. The final component was the plus modifier, first proposed in [50]. The algebraic properties of these schemes were intended to complement their weaknesses when used in conjunction. This patchwork design requires, however, a careful consideration of the relevant cryptanalyses within all of these families.

The original oil-and-vinegar (OV) scheme, proposed in [46], was completely broken in [61] by what we call the invariant method. Specifically, the balanced OV scheme contains an equal number of oil variables, variables which only occur linearly in the central map, and vinegar variables, which occur quadratically. Thus, the differential of any central polynomial has the shape

$$Df_i = \begin{bmatrix} a_{1,1} & \cdots & a_{1,v} & a_{1,v+1} & \cdots & a_{1,2v} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{1,v} & \cdots & a_{v,v} & a_{v,v+1} & \cdots & a_{v,2v} \\ a_{1,v+1} & \cdots & a_{v,v+1} & 0 & \cdots & 0 \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{1,2v} & \cdots & a_{v,2v} & 0 & \cdots & 0 \end{bmatrix},$$

under an appropriate basis of $\mathbb{F}^{2v} = V \oplus O$, where V is the subspace spanned by the vinegar variables and O is the subspace spanned by the oil variables.

The invariant attack proceeds by computing the differential of random linear combinations of the public polynomials until two full rank differentials, Df_1 and Df_2, are produced. Then O is left invariant by $Df_1^{-1}Df_2$ and is thus easily recovered. A similar technique has been used in conjunction with rank attacks to assault schemes with a similar structure whenever $\dim(V) \leq \dim(O)$, see, in particular, [39, 40, 41].

HFE and some of its modifications have been the target of effective cryptanalyses utilizing the low Q-rank property of the central map. Each of these cryptanalyses can be described as a big field MinRank attack, recovering a low rank quadratic form over the extension $\mathbb{E}$ from which an isomorphism relating the public key to an equivalent private key can be derived.

The earliest iteration of this technique is the well-known Kipnis-Shamir (KS) attack of [34], also known by the name MinRank, due to the close relationship between the attack and the MinRank problem in algebraic complexity theory, see

[23]. The KS-attack recovers a private key for HFE by exploiting the fact that the low Q-rank of the central map is a property preserved by isomorphisms. Considering an odd characteristic instance of HFE. We may write the homogeneous quadratic part of the central map as

$$\begin{bmatrix} x & x^q & \cdots & x^{q^{n-1}} \end{bmatrix} \begin{bmatrix} \alpha_{0,0} & \alpha'_{0,1} & \cdots & \alpha'_{0,d-1} & 0 & \cdots & 0 \\ \alpha'_{0,1} & \alpha_{1,1} & \cdots & \alpha'_{1,d-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \alpha'_{0,d-1} & \alpha'_{1,d-1} & \cdots & \alpha_{d-1,d-1} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & 0 & \cdots & 0 \end{bmatrix} \begin{bmatrix} x \\ x^q \\ \vdots \\ x^{q^{n-1}} \end{bmatrix},$$

where $\alpha'_{i,j} = \frac{1}{2}\alpha_{i,j}$ and $d = \lceil \log_q(D) \rceil$. The KS-attack first interpolates an univariate representation of the public key over $\mathbb{E}$. This representation of the public key is isomorphic to the central map of Q-rank bounded by the ceiling of the logarithm of the degree bound. Thus, there is a linear map T^{-1} which when composed with the public key has Q-rank d, and so there is a low rank matrix that is an $\mathbb{E}$-linear combination of the Frobenius powers of G. This turns recovery of the transformation T into the solution of a MinRank problem over $\mathbb{E}$.

Another version of this attack, utilizing the same property, is the key recovery attack of [6]. The authors prove the existence of an $\mathbb{E}$-linear combination of the *public* key with low rank over $\mathbb{E}$. Setting the unknown coefficients of this linear combination as variables, they construct the ideal $I \subseteq R = \mathbb{F}[T]$ of minors of this sum of the appropriate dimension such that $V(I) \cap \mathbb{E}^{\dim(R)}$ consists of exactly such linear coefficients. Thus a Gröbner basis needs to be computed over $\mathbb{F}$ and the variety computed over $\mathbb{E}$. This modeling of the KS-attack is called minors modeling and dramatically improves the efficiency of the KS-attack in many circumstances.

The KS-attack with either KS modeling or with minors modeling has also been used to break other HFE descendants. In [6], the minors modeling approach is used to break multi-HFE. In [67], the KS-attack is extended to provide key recovery for HFE-. In [10], both the KS modeling and minors modeling versions of the KS-attack are used to undermine the security of ZHFE.

The MinRank methodology is also employed in [52], where an effective key recovery attack on SRP is presented. It was shown that the low Q-rank of Square is exposed by the SRP construction. Specifically, the Q-rank of the square map $f(x) = x^2$ is one over an odd characteristic field. Since this low Q-rank map is in the span of the public polynomials, there is an $\mathbb{E}$-linear combination of the public polynomials of rank one! Thus the ideal generated by the two-by-two minors is resolved at degree two and the complexity of the attack is $\mathcal{O}(\binom{m+1}{2}^{\omega})$, where $2 \leq \omega \leq 3$ is the linear algebra constant. The attack is applied practically, breaking the 80-bit parameters in about 8 minutes.

5.4 New Scheme - HFERP

In this section, we present a significant modification of SRP that we call HFERP. The key observation is that by replacing the Square map with a higher Q-rank instance of HFE, one can make the MinRank attack inefficient while maintaining efficient inversion. For simplicity of the exposition, we present the scheme with a single layer UOV component, noting that it is trivial to replace UOV with a multi-layer Rainbow via the same construction.

Choose a finite field $\mathbb{F}_q$ and let $\mathbb{E}$ be a degree d extension field over $\mathbb{F}_q$. Let $\phi : \mathbb{F}_q^d \to \mathbb{E}$ be an $\mathbb{F}_q$-vector space isomorphism. Also, let $o, r, s,$ and l be non-negative integers.

Key Generation Let $n = d + o - l$, $n' = d + o$ and $m = d + o + r + s$. The central map of HFERP is the concatenation of an HFE core map, $\mathcal{F}_{HFE}$, an UOV (or alternatively, Rainbow) section, $\mathcal{F}_R$, and the plus modifier, $\mathcal{F}_P$. Formal definitions of the maps are provided below:

- The HFE component is defined as $\mathcal{F}_{HFE} : \mathbb{F}_q^{n'} \to \mathbb{F}_q^d$ and is the result of the following composition:

  $$\mathbb{F}_q^{n'} \xrightarrow{\pi_d} \mathbb{F}_q^d \xrightarrow{\phi} \mathbb{E} \xrightarrow{f} \mathbb{E} \xrightarrow{\phi^{-1}} \mathbb{F}_q^d$$

  where f is the HFE core map described in (3.1) and $\pi_d : \mathbb{F}_q^{d+o} \to \mathbb{F}_q^d$ is the projection onto the first d coordinates.

- The UOV (or alternatively, Rainbow) component is defined as

  $$\mathcal{F}_R = (g^{(1)}, \ldots, g^{(o+r)}) : \mathbb{F}_q^{n'} \to \mathbb{F}_q^{o+r}$$

  following the normal construction of the UOV signature scheme where $\mathcal{V} = \{1, \ldots, d\}$ and $\mathcal{O} = \{d+1, \ldots, d+o\}$. For every $k \in \{1, \ldots, o+r\}$, the quadratic polynomial $g^{(k)}$ is of the following form:

  $$g^{(k)}(x_1, \ldots, x_{n'}) = \sum_{i \in \mathcal{O}, j \in \mathcal{V}} \alpha^{(k)} x_i x_j + \sum_{i,j \in \mathcal{V}, i \leq j} \beta_{i,j}^{(k)} x_i x_j + \sum_{i \in \mathcal{V} \cup \mathcal{O}} \gamma_i^{(k)} x_i + \eta^{(k)}$$

  where $\alpha^{(k)}$, $\beta_{i,j}^{(k)}$, $\gamma_i^{(k)}$, and $\eta^{(k)}$ are chosen at random from $\mathbb{F}_q$.

- The Plus modification is defined as $\mathcal{F}_P = (h^{(1)}, \ldots, h^{(s)}) : \mathbb{F}_q^{n'} \to \mathbb{F}_q^s$ which consists of s randomly generated quadratic polynomials.

An affine embedding $\mathcal{U} : \mathbb{F}_q^n \to \mathbb{F}_q^{n'}$ of full rank and an affine isomorphism $\mathcal{T} : \mathbb{F}_q^m \to \mathbb{F}_q^m$ are chosen for the butterfly construction as is common in big field schemes. The public key is given by $\mathcal{P} = \mathcal{T} \circ \mathcal{F} \circ \mathcal{U} : \mathbb{F}_q^n \to \mathbb{F}_q^m$, where $\mathcal{F} = \mathcal{F}_{HFE} \| \mathcal{F}_R \| \mathcal{F}_P$ ($\|$ being the concatination function), and the private key is represented by the following figure:

$$\begin{tikzcd}
& & \mathbb{F}_q^d \arrow[dr] & & \\
\mathbb{F}_q^n \arrow[r, "\mathcal{U}"] \arrow[rrrr, bend right=50, "\mathcal{P}"'] & \mathbb{F}_q^{n+l} \arrow[ur, "\mathcal{F}_{HFE}"] \arrow[r, "\mathcal{F}_R"] \arrow[dr, "\mathcal{F}_P"'] & \mathbb{F}_q^{o+r} \arrow[r] & \mathbb{F}_q^m \arrow[r, "\mathcal{T}"] & \mathbb{F}_q^m \\
& & \mathbb{F}_q^s \arrow[ur] & &
\end{tikzcd}$$

Encryption Given a message $M \in \mathbb{F}_q^n$, the ciphertext is computed as $C = \mathcal{P}(M) \in \mathbb{F}_q^m$.

Decryption Given a ciphertext $C = (c_1, \ldots, c_m) \in \mathbb{F}_q^m$, the decryption process is the following:

1. Compute $\mathbf{x} = (x_1, \ldots, x_m) = \mathcal{T}^{-1}(C)$.

2. Compute $\mathbf{X} = \phi(x_1, \ldots, x_d) \in \mathbb{E}$.

3. Use the Berlekamp algorithm to compute the inverse of the HFE polynomials to recover $\mathbf{y} = (y_1, \ldots, y_d)$.

4. Given the vinegar values $y_1, \ldots, y_d$, solve the system of $o + r$ linear equations in the $n' - d = o$ variables $u_{d+1}, \ldots, u_{n'}$ given by

$$g^{(k)}(y_1, \ldots, y_d, u_{d+1}, \ldots, u_{n'}) = x_{d+k}$$

   for $k = 1, \ldots, o + r$. The solution is denoted $(y_{d+1}, \ldots, y_{n'})$.

5. Compute the plaintext $M \in \mathbb{F}_q^n$ by finding the preimage of $(y_1, \ldots, y_{n'})$ under the affine embedding $\mathcal{U}$.

5.5 Complexity of Known Attacks

In this section we derive tight complexity estimates or proofs of resistance for the principal relevant attacks on HFERP. These attacks include the direct algebraic attack, the MinRank attack, the small field MinRank and dual rank attacks, and the invariant attack.

5.5.1 Algebraic Attack

The algebraic attack attempts to invert the public key at a ciphertext directly via the calculation of a Gröbner basis. It is commonly believed that the closeness of the solving degree of a polynomial system, the degree at which the Gröbner basis is resolved, and the degree of regularity, the degree at which a non-trivial syzygy producing a degree fall first occurs, is a generic property. Thus the lower bound on the complexity of the algebraic attack that the degree of regularity provides is likely a tight bound, and is consequently a critical quantity for analyzing the security of the scheme.

Theorem 5.1 (see [30]). *The degree of regularity of the public key of HFERP is bounded by*

$$d_{reg} \leq \begin{cases} \frac{(q-1)\lceil \log_q(D) \rceil}{2} + 2 \text{ \textit{if} } q \text{ \textit{is odd or} } \lceil \log_q(D) \rceil \text{ \textit{is even,}} \\ \frac{(q-1)\left(\lceil \log_q(D) \rceil + 1\right)}{2} + 1 \text{ \textit{otherwise.}} \end{cases}$$

Proof. There is a linear function of the public key separating the HFE polynomials $\mathcal{H}$ from the non-HFE polynomials $\mathcal{N}$. Trivially, the d_{reg} is bounded by the degree of regularity of the system $\mathcal{H}$, which, via Theorem 4.2 in [17], produces the above bound. □

One must note that the above bound is not what is needed to ensure security. Instead we require a lower bound. Extensive experimentation shows that for very small q, the above estimate is tight. We have, however, a further complication. In general, adding more polynomials to an ideal may decrease its degree of regularity. To address this issue we have conducted small scale experiments showing that the degree of regularity and solving degree behave similarly to those of random systems, see Section 5.6.

Conjecture 5.1 (see [30]). *Under the assumption that the degree of regularity is at least $\lceil \log_q(D) \rceil + 2$ for small odd q and sufficiently large n, the complexity of the algebraic attack is given by*

$$Comp._{alg} = \mathcal{O}\left(\binom{n + d_{reg}}{d_{reg}}^2 \binom{n}{2} \right) = \mathcal{O}\left(n^{2\lceil \log_q(D) \rceil + 6} \right).$$

5.5.2 MinRank Attack

The min-rank attack proposed in [52] is so successful due to the Q-rank of the squaring map within SRP being equal to one. By changing the square map component to an HFE core map, we are able to thwart such an attack on HFERP. This subsection walks through the attack proposed in [52] , with HFERP in mind, and proves that the min-Q-rank of HFERP differs from SRP.

Note that, similar to SRP, the public key of HFERP has an analogous scheme without embedding as long as $\pi_d \circ \mathcal{U}$ is of full rank, which it is defined to be in this scheme. Let $\pi'_d : \mathbb{F}_q^n \to \mathbb{F}_q^d$ be the projection onto the first d coordinates and find a projection $\rho : \mathbb{F}_q^{n+l} \to \mathbb{F}_q^n$ such that $\mathcal{U}' = \rho \circ \mathcal{U}$ has full rank and $\pi'_d \circ \mathcal{U}' = \pi_d \circ \mathcal{U}$. Let $\mathcal{F}^* : \mathbb{E} \to \mathbb{E}$ represent the chosen high Q-rank HFE core map so that $\mathcal{F}_{HFE} = \phi^{-1} \circ \mathcal{F}^* \circ \phi \circ \pi_d$. Then identify the Rainbow and random components as $\mathcal{F}'_R : \mathcal{F}_R \circ \mathcal{U} \circ \mathcal{U}'^{-1}$ and $\mathcal{F}'_P : \mathcal{F}_P \circ \mathcal{U} \circ \mathcal{U}'^{-1}$ respectively. Thus, one can see that

$$\mathcal{T} \circ \begin{bmatrix} \phi \circ \mathcal{F}^* \circ \phi^{-1} \circ \pi_d \\ \mathcal{F}_R \\ \mathcal{F}_P \end{bmatrix} \circ \mathcal{U} = \mathcal{T} \circ \begin{bmatrix} \phi \circ \mathcal{F}^* \circ \phi^{-1} \circ \pi'_d \\ \mathcal{F}'_R \\ \mathcal{F}'_P \end{bmatrix} \circ \mathcal{U}'.$$

Notice that the attack on SRP was not just a min-rank attack on the public key of SRP, but on a linear combination of public forms of SRP that had low Q-rank over the degree d extension used by the squaring component. This method allowed the attack to ignore the fact that the public key of an instance of SRP was expected to be of high rank. Thus, to demonstrate that HFERP resists such an attack, we briefly outline the method of deriving the linear combination of public forms from [52] for HFERP and prove that the min-Q-Rank of the result is sufficiently high to resist such an attack.

Let α be a primitive element of the degree d extension $\mathbb{E}$ of $\mathbb{F}_q$. Fix a vector space isomorphism $\phi : \mathbb{F}_q^d \to \mathbb{E}$ defined by $\phi(\bar{x}) = \sum_{i=0}^{d-1} x_i \alpha^i$. Then, fix a one dimensional representation $\Phi : \mathbb{E} \to \mathbb{A}$ defined by $a \xrightarrow{\Phi} (a, a^q, \dots, a^{q^{d-1}})$. Next, define $\mathcal{M}_d : \mathbb{F}_q^d \to \mathbb{A}$ by $\mathcal{M}_d = \Phi \circ \phi$. It was demonstrated you can look at this map through the following matrix representation

$$\mathbf{M}_d = \begin{bmatrix} 1 & 1 & \dots & 1 \\ \alpha & \alpha^q & \dots & \alpha^{q^{d-1}} \\ \alpha^2 & \alpha^{2q} & \dots & \alpha^{2q^{d-1}} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{d-1} & \alpha^{(d-1)q} & \dots & \alpha^{(d-1)q^{d-1}} \end{bmatrix} \in \mathcal{M}_{d\times d}(\mathbb{E})$$

This matrix allows the passage from $\mathbb{F}_q^d$ and $\mathbb{A}$ easily by right multiplication with $\mathbf{M}_d$ or $\mathbf{M}_d^{-1}$. Next are a few more definitions necessary to be able to look at a matrix representation of the public key:

$$\widetilde{\mathbf{M}}_d = \begin{bmatrix} \mathbf{M}_d & 0 \\ 0 & \mathbf{I}_{o+r+s} \end{bmatrix} \in \mathcal{M}_{m\times m}(\mathbb{E})$$

$$\widehat{\mathbf{M}}_d = \begin{bmatrix} \mathbf{M}_d \\ \mathbf{0}_{o\times d} \end{bmatrix} \in \mathcal{M}_{(d+o)\times d}(\mathbb{E})$$

Finally, define $\mathbf{F}^{*i}$ be the matrix representation of the quadratic form over $\mathbb{A}$ of the i^{th} Frobenius power of the chosen HFE core map. Now we have all the necessary notation to view the public key as a matrix equation.

Denote the m-dimensional vector of $(d+o)\times(d+o)$ symmetric matrices associated by the private key as follows:

$$(\mathbf{F}_{(HFE,0)},\dots,\mathbf{F}_{(HFE,d-1)},\mathbf{F}_{(R,0)},\dots,\mathbf{F}_{(R,o+r-1)}\mathbf{F}_{(P,0)},\dots,\mathbf{F}_{(P,s-1)}). \tag{5.1}$$

Note that the function corresponding to the application of each coordinate of a vector of the quadratic forms followed by the application of a linear map represented by a matrix is denoted as a right product of the vector and a matrix representation of the linear map.

Next, observe

$$(\mathbf{F}_{(HFE,0)},\dots,\mathbf{F}_{(HFE,d-1)})\mathbf{M}_d = (\widehat{\mathbf{M}}_d\mathbf{F}^{*0}\widehat{\mathbf{M}}_d^{\top},\dots,\widehat{\mathbf{M}}_d\mathbf{F}^{*(d-1)}\widehat{\mathbf{M}}_d^{\top}),$$

which yields

$$\begin{aligned}&(\bar{x}\mathbf{F}_{(HFE,0)}\bar{x}^{\top},\dots,\bar{x}\mathbf{F}_{(HFE,d-1)}\bar{x}^{\top})\mathbf{M}_d =\\ &\qquad\qquad(\bar{x}\widehat{\mathbf{M}}_d\mathbf{F}^{*0}\widehat{\mathbf{M}}_d^{\top}\bar{x}^{\top},\dots,\bar{x}\widehat{\mathbf{M}}_d\mathbf{F}^{*(d-1)}\widehat{\mathbf{M}}_d^{\top}\bar{x}^{\top}),\end{aligned}$$

as a function of $\bar{x}$. This gives the following equation:

$$\begin{aligned}(\mathbf{F}_{(HFE,0)},\dots,&\mathbf{F}_{(HFE,d-1)},\mathbf{F}_{(R,0)},\dots,\mathbf{F}_{(P,s-1)})\widetilde{\mathbf{M}}_d =\\ &(\widehat{\mathbf{M}}_d\mathbf{F}^{*0}\widehat{\mathbf{M}}_d^{\top},\dots,\widehat{\mathbf{M}}_d\mathbf{F}^{*(d-1)}\widehat{\mathbf{M}}_d^{\top},\mathbf{F}_{(R,0)},\dots,\mathbf{F}_{(P,s-1)})\end{aligned} \tag{5.2}$$

Now, look to the relation between the public key and its corresponding private key central maps:

$$(\mathbf{P}_0,\dots,\mathbf{P}_{m-1})\mathbf{T}^{-1} = (\mathbf{U}\mathbf{F}_{(HFE,0)}\mathbf{U}^{\top},\dots,\mathbf{U}\mathbf{F}_{(P,s-1)}\mathbf{U}^{\top}). \tag{5.3}$$

By combining equations 5.2 and 5.3, we have the following:

$$\begin{aligned}&(\mathbf{P}_0,\dots,\mathbf{P}_{m-1})\mathbf{T}^{-1}\widetilde{\mathbf{M}}_d =\\ &(\mathbf{U}\widehat{\mathbf{M}}_d\mathbf{F}^{*0}\widehat{\mathbf{M}}_d^{\top}\mathbf{U}^{\top},\dots,\mathbf{U}\widehat{\mathbf{M}}_d\mathbf{F}^{*(d-1)}\widehat{\mathbf{M}}_d^{\top}\mathbf{U}^{\top},\mathbf{U}\mathbf{F}_{(R,0)}\mathbf{U}^{\top},\dots,\mathbf{U}\mathbf{F}_{(P,s-1)}\mathbf{U}^{\top})\end{aligned}$$

As in [52], let $\widehat{\mathbf{T}} = \mathbf{T}^{-1}\widehat{\mathbf{M}}_d = [t_{i,j}] \in \mathcal{M}_{m\times m}(\mathbb{E})$ and $\mathbf{W} = \mathbf{U}\widehat{\mathbf{M}}_d$. This identification produces

$$\sum_{i=0}^{m-1} t_{i,0}\mathbf{P}_i = \mathbf{W}\mathbf{F}^{*0}\mathbf{W}^\intercal. \tag{5.4}$$

Since the rank of $\mathbf{F}^{*i}$ is equal to the Q-rank of the quadratic form of the HFE core map for all i, the rank of this $\mathbb{E}$-linear combination of the public matrices is bounded by the minimum of the rank of $\mathbf{U}\widehat{\mathbf{M}}_d$ and the rank of $\mathbf{F}^{*0}$, *id est* the Q-rank of our HFE core map. This statement forms the following theorem:

Theorem 5.2 (see [30]). *The min-Q-rank of the public key P of HFERP(q,d,o,r,s,l) is given by:*

$$\text{min-Q-rank(P)} \leq \min\{Rank(\mathbf{U}\widehat{\mathbf{M}}_d), Rank(\mathbf{F}^{*0})\}$$

Proof. The proof in [52] describes the parameters in which the min-Q-rank(P) can be equal to zero. So, we move forward with the assumption that $\mathbf{U}\widehat{\mathbf{M}}_d \neq 0$, which occurs with high probability when $d > l$. In (5.4) we have a linear combination of the public key equations equal to the following:

$$\mathbf{W}\mathbf{F}^{*0}\mathbf{W}^\intercal = \mathbf{U}\widehat{\mathbf{M}}_d\mathbf{F}^{*0}\widehat{\mathbf{M}}_d^\intercal\mathbf{U}^\intercal. \tag{5.5}$$

This proves our result. □

It should be noted that $\mathbf{U}$, $\widehat{\mathbf{M}}_d$, and $\mathbf{F}^{*0}$ are chosen by the user. They can easily be chosen in such a way such that

$$\text{min-Q-rank(P)} = \min\{Rank(\mathbf{U}\widehat{\mathbf{M}}_d), Rank(\mathbf{F}^{*0})\}.$$

This would also occur with high probability if $\mathbf{U}$, $\widehat{\mathbf{M}}_d$, and $\mathbf{F}^{*0}$ were randomly generated. Directly from [67], we also have the following complexity for the MinRank attack on HFERP:

Corollary 5.1 (see [30]). *The complexity of the MinRank attack with minors modeling on HFERP is given by*

$$Comp._{Minors} = \mathcal{O}\left(\binom{m+\lfloor \log_q(D)\rfloor}{\lceil \log_q(D)\rceil}^2 \binom{m}{2}\right) = \mathcal{O}\left(m^{2\lceil \log_q(D)\rceil+2}\right).$$

5.5.3 Base-Field Rank and Invariant Attacks

Variants of several attacks applicable to other versions of the Rainbow cryptosystem are applicable to HFERP. These include the linear-algebra-search version of MinRank [28], the HighRank attack [28] and the UOV invariant attack [32].

The MinRank attack works by randomly choosing one or more vectors $\mathbf{w}_j$ in the plaintext space and solving for a linear combination $t_i \in \mathbb{F}$ of the plaintext equations satisfying:

$$\sum_{i=1}^{m} t_i Df_i(\mathbf{w}_j) = 0$$

The attack succeeds when $\mathbf{w}_j$ is in the kernel of a low rank linear combination of differentials of the public polynomials. In the case of HFERP, the HFE component equations form a d-dimensional subspace of the public equations having rank d over $\mathbb{F}$. Note that the attacker can remove up to $d-1$ equations while preserving at least a one dimensional subspace of low rank maps. Thus, the attack can succeed with a one dimensional solution space for t_i and only a single $\mathbf{w}_j$ as long as $m \leq n+d$.

If $m > n+d$, the adversary may still use a single vector $\mathbf{w}_j$ to constrain the t_i's rather than attempting to find two vectors in the kernel of the HFE equations. In this case, the attacker must search through an $m-n-d+1$ dimensional space of spurious solutions to find the useful 1 dimensional space of t_is. This method is still less expensive than searching for two vectors in the kernel of the HFE equations when $m < n+2d$.

It should be further noted that, since the differentials of the oil maps will map any vector in the kernel of the HFE equations to the d-dimensional HFE input space, we expect an $o_1 + r_1 - d$ dimensional subspace of the oil equations to also have such a vector in the kernel of their differentials, see Figure 5.1. Thus, when $m < n + \max(d, o_1 + r_1)$, vectors in the HFE kernel can be recognized, because they are in the kernel of an unusually large subspace of the public equations, and when $2d < n$ the linear combinations of the public equations from the HFE and oil spaces can be recognized due to their low rank.

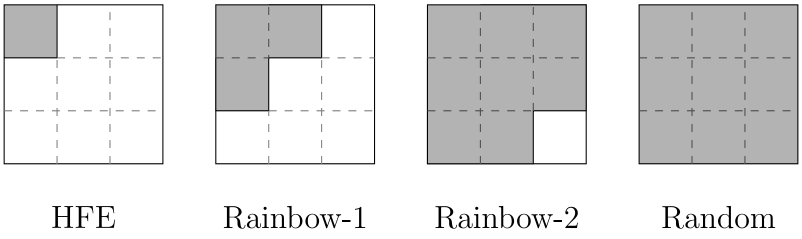


Figure 5.1: The shape of the matrix representations of the central maps of HFERP. The shaded regions represent possibly nonzero values while unshaded areas have coefficients of zero.

Thus the complexity of MinRank (for plausible choices of m) is

$$Comp._{MinRank} = \begin{cases} \mathcal{O}\left(q^{d}m^{\omega}\right) & m < n + \max(d, o_1 + r_1) \\ \\ \mathcal{O}\left(q^{d+m-n-\max(d,o_1+r_1)}n^{\omega}\right) & m \geq n + \max(d, o_1 + r_1) \\ & m < n + d + \max(d, o_1 + r_1) \\ & n > 2d \\ \\ \mathcal{O}\left(q^{m-n}n^{\omega}\right) & m \geq n + \max(d, o_1 + r_1) \\ & m < n + 2d \\ & n \leq 2d \\ \\ \mathcal{O}\left(q^{2d}m^{\omega}\right) & m < 2n + \max(d, o_1 + r_1 - d) \\ & \text{No better attack.} \end{cases}$$

In the HighRank attack, the attacker randomly selects linear combinations of the public polynomials with the hope of selecting a polynomial with significantly less than full rank. This attack takes advantage of the $d + o_1 + r_1$-dimensional subspace of the public polynomials generated by the HFE maps and either the Rainbow-1 maps of Figure 5.1 or for UOV of the d-dimensional HFE subspace. The complexity of the attack is then:

$$Comp._{HighRank} = \mathcal{O}\left(q^{m-d-o_1-r_1}n^{\omega}\right).$$

It should also be noted that linear combinations of HFE and Rainbow-1 polynomials form an $m - s$ dimensional subspace of the public polynomials, that act linearly on the $o_2 - l$ dimensional preimage under $\mathcal{U}$ of the oil subspace. This bounds their rank to be at most $2d$. Noting that the probability that a random square matrix has corank a is approximately q^{-a^2}, we see that, the high rank attack can be straightforwardly applied if $2d < n - \sqrt{m-d-o_1-r_1}$.

Additionally, the HighRank attack can be combined with the oil and vinegar invariant attack to distinguish linear combinations of the HFE and Rainbow maps from other linear combinations of the public maps. Here, a pair of maps from the HFE and Rainbow subspace can be identified by restricting their differentials to a subspace of the plaintext space in which both maps are full rank, and checking to see if $(Dp_1)^{-1}Dp_2$ has a large invariant subspace (which will be the intersection of the preimage of the oil subspace under $\mathcal{U}$ and the subspace used to restrict the differentials). This allows the high rank attack to be applied with similar complexity as long as $2d < n - \sqrt{\frac{m-d-o_1-r_1}{2}}$: Applying the attack will involve testing no more than $\left(q^{\frac{m-d-o_1-r_1}{2}}\right)^2 = q^{m-d-o_1-r_1}$ pairs of rank $n-2d$ maps, and therefore this step will not dominate the complexity of the approximately $q^{m-d-o_1-r_1}$ rank computations involved in the HighRank step.

If $2d \geq \zeta$, where $\zeta_1 = n - \sqrt{\frac{m-d-o_1-r_1}{2}}$, the complexity of HighRank is given by:

$$Comp._{HighRank} = \begin{cases} Comp._{HighRank} = \mathcal{O}\left(q^{m-d}n^{\omega}\right) & 2d \geq \zeta_1 \\ Comp._{HighRank} = \mathcal{O}\left(q^{m-d-o_1-r_1}n^{\omega}\right) & 2d < \zeta_1. \end{cases}$$

Finally, when $2d \geq n - \sqrt{\frac{m-d-o_1-r_1}{2}}$, as in the UOV attack, the previous steps must be combined with a projection, aimed at removing enough vinegar variables that the restriction of the differentials of linear combinations of HFE and Rainbow maps to the projected plaintext space is less than full rank. This yields a complexity for hybrid HighRank/UOV invariant type attacks of:

$$Comp._{UOV} = \begin{cases} \mathcal{O}\left(q^{m-d-o_1-r_1}n^{\omega}\right) & n > \zeta_2 \\ \mathcal{O}\left(q^{m-d-o_1-r_1+\sqrt{\frac{m-d-o_1-r_1}{2}}+2d-n}(o_1+o_2-l)^4\right) & n \leq \zeta_2. \end{cases}$$

where $\zeta_2 = 2d + \sqrt{\frac{m-d-o_1-r_1}{2}}$. This attack may also be applied to the Rainbow-2 maps of Figure 5.1 in which case the complexity is:

$$Comp._{UOV2} = \begin{cases} \mathcal{O}\left(q^s n^\omega\right) & n > 2d + 2o_1 + \sqrt{\frac{s}{2}} \\ \mathcal{O}\left(q^{s+\sqrt{\frac{s}{2}}+2d+2o_1-n}(o_2 - l)^4\right) & n \leq 2d + 2o_1 + \sqrt{\frac{s}{2}}. \end{cases}$$

5.6 Parameter Selection and Experimental Results

We propose single-layer parameters (A) and (B) for 80-bit security and multi-layer parameters (C) and (D) for 128-bit security :

(A) $(q = 3, d = 42, o = 21, r = 15, s = 17, l = 0, D = 3^7 + 1)$

(B) $(q = 3, d = 63, o = 21, r = 11, s = 10, l = 0, D = 3^7 + 1)$

(C) $(q = 3, d = 85, o_1 = o_2 = 70, r_1 = r_2 = 89, s = 61, l = 0, D = 3^7 + 1)$

(D) $(q = 3, d = 60, o_1 = o_2 = 40, r_1 = r_2 = 23, s = 40, l = 0, D = 3^9 + 1)$

Then we have the following values for (n, m): $(63, 95)$ for (A), $(84, 105)$ for (B), $(225, 464)$ for (C), and $(140, 226)$ for (D). The security level for suggested parameters is estimated by all the attack in §6. Here, we assume that the degree of regularity for direct attack is 10 by Conjecture 1 for (A),(B), and (C) while it is 12 for (D).

To draw a direct comparison with HFE, note that to achieve the same security level as HFERP, an HFE scheme requires m equations, and hence $n = m$ variables. Therefore secure HFE public keys are far larger while offering slower decryption due to the use of the Berlekamp algorithm in a far larger field.

We ran a series of experiments with Magma, see [8], on a 2.6 GHz Intel® Xeon$^{\mathrm{R}}$ CPU. These are not optimized implementations.

	(A)	(B)	(C)	(D)
Key Generation	0.299 s	0.572 s	20.498 s	3.43 s
Encryption	0.001 s	0.001 s	0.006 s	0.001 s
Decryption	3.977 s	8.671 s	49.182 s	124.27 s
Secret Key Size	19.8KB	31.7KB	1344.0KB	226.0KB
Public Key Size	48.2KB	93.6KB	2905.7KB	552.3KB

Table 5.1: Experimental results for HFERP.

We also investigated the growth of the first fall degree (d_{reg}) as well as the solving degree with five experiments performed at each of eight different parameters sets. We directly compared these data with randomly generated systems, see Table 5.2.

For comparison, we include the semi-regular degree for systems of m equations in n variables. This quantity was calculated by computing the first non-positive coefficient in the series

$$S_{n,m}(t) = \frac{(1-t^q)^n(1-t^2)^m}{(1-t)^n(1-t^{2q})^m}.$$

Noting that the degree of regularity of the zero-dimensional ideal is the same as the first fall degree of the ideal generated by the homogeneous components of the generators of highest degree. We derive the above formula as the fusion of the techniques in [69] and [1].

It is clear that the degree of regularity of the small scale instances of HFERP grows in relation to that of random schemes. By the data in the tables, we can estimate that the degree of regularity for direct attack on (A) and (B) is greater than 9 at least.

Table 5.2: Direct attack experiment data for various values of d, o, r, s. (s.r.d. stands for semi- regular degree)

			HFERP		Random		
(q, d, o, r, s, l, D)	n	m	d_{reg}	sol. deg	d_{reg}	sol. deg	s.r.d.
$(3, 8, 4, 3, 3, 0, 2188)$	12	18	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	4
$(3, 10, 5, 4, 3, 0, 2188)$	15	22	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	5
$(3, 12, 6, 5, 4, 0, 2188)$	18	27	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	5
$(3, 14, 7, 5, 5, 0, 2188)$	21	31	$6, 5, 5, 5, 5$	$6, 6, 6, 6, 6$	$5, 5, 5, 5, 5$	$6, 6, 6, 6, 6$	6

Table 2.A. Direct Attack, $d = 2o, d + o \fallingdotseq 2(r + s), o = 4, 5, 6, 7$

			HFERP		Random		
(q, d, o, r, s, l, D)	n	m	d_{reg}	sol. deg	d_{reg}	sol. deg	s.r.d.
$(3, 9, 3, 2, 2, 0, 2188)$	12	16	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	5
$(3, 12, 4, 2, 2, 0, 2188)$	16	20	$5, 6, 6, 5, 5,$	$5, 6, 6, 6, 5$	$6, 5, 6, 6, 5$	$6, 6, 6, 6, 6$	6
$(3, 15, 5, 3, 3, 0, 2188)$	20	26	$6, 5, 5, 5, 5$	$6, 6, 6, 6, 6$	$5, 5, 5, 6, 5$	$6, 6, 6, 6, 6$	6
$(3, 18, 6, 3, 3, 0, 2188)$	24	30	$5, 5, 5, 5, 5$	$7, 7, 7, 7, 7$	$5, 5, 5, 5, 7$	$7, 7, 7, 7, 7$	7

Table 2.B. Direct Attack, $d = 3o, r + s \fallingdotseq o, o = 3, 4, 5, 6$

			HFERP		Random		
(d, o, r, s, l, D)	n	m	d_{reg}	sol. deg	d_{reg}	sol. deg	s.r.d.
$(3, (3, 3), (4, 4), 2, 0, 2188)$	9	19	$3, 3, 3, 3, 3$	$3, 3, 2, 3, 2$	$3, 3, 3, 3, 3$	$2, 3, 3, 2, 2$	3
$(7, (6, 6), (7, 7), 5, 0, 2188)$	19	38	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	5
$(10, (8, 8), (11, 11), 7, 0, 2188)$	26	55	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	5
$(14, (11, 11), (14, 14), 10, 0, 2188)$	36	74	5		5		6

Table 2.C. Direct Attack, $d \fallingdotseq 3.4a, o \fallingdotseq (2.8a, 2.8a), r \fallingdotseq (3.56a, 3.56a), s \fallingdotseq 2.44a, a = 1, 2, 3, 4$

			HFERP		Random		
(d, o, r, s, l, D)	n	m	d_{reg}	sol. deg	d_{reg}	sol. deg	s.r.d.
$(5, (3, 3), (2, 2), 3, 0, 3^9 + 1)$	11	18	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	$4, 4, 4, 3, 4$	4
$(7, (5, 5), (3, 3), 5, 0, 3^9 + 1)$	17	28	$4, 4, 4, 4, 4$	$4, 4, 4, 4, 4$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	5
$(10, (6, 6), (4, 4), 6, 0, 3^9 + 1)$	22	36	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$5, 5, 5, 5, 5$	$6, 6, 6, 6, 6$	6
$(12, (8, 8), (5, 5), 8, 0, 3^9 + 1)$	28	46	$5, 5$	$6, 6$	$5, 5$	6	6

Table 2.D. Direct Attack, $d \fallingdotseq 2.4a, o \fallingdotseq (1.6a, 1.6a), r \fallingdotseq (0.92a, 0.92a), s \fallingdotseq 1.6a, a = 2, 3, 4, 5$

5.7 Toy Example

The purpose of the following toy example is to help the reader understand the process of generating a public key for an instance of HFERP as well as an example of encryption and decryption. The parameters used are by no means secure and are soley for instructional purposes.

Parameters of this toy example are as follows: $q = 7$, $d = o = r = 2$, $s = 1$, and $l = 0$. Then, construct $\mathbb{E}$ a degree 2 extension field over $\mathbb{F}_7$. The chosen HFE core map is $f = \xi^{12}X^{14} + \xi^6X^8 + \xi^{29}X^2$ where $\xi \in \mathbb{E}$. Let $\mathcal{T}$ and $\mathcal{U}$ be the following affine maps:

$$\mathcal{T} = \begin{bmatrix} 2 & 1 & 2 & 4 & 5 & 0 & 3 \\ 1 & 1 & 3 & 3 & 4 & 4 & 4 \\ 4 & 2 & 1 & 3 & 1 & 0 & 6 \\ 0 & 1 & 0 & 1 & 5 & 5 & 5 \\ 5 & 5 & 3 & 6 & 4 & 2 & 4 \\ 2 & 5 & 1 & 6 & 5 & 6 & 0 \\ 1 & 1 & 2 & 2 & 6 & 4 & 3 \end{bmatrix}, \mathcal{U} = \begin{bmatrix} 4 & 6 & 6 & 4 \\ 3 & 2 & 0 & 2 \\ 1 & 1 & 6 & 5 \\ 3 & 6 & 6 & 6 \end{bmatrix}$$

With the parameters described above, $\mathcal{F}$ can be represented as the follwing matrices over $\mathbb{F}_7$

$$F_1 = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, F_2 = \begin{bmatrix} 0 & 3 & 0 & 0 \\ 1 & 6 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}, F_3 = \begin{bmatrix} 3 & 1 & 6 & 1 \\ 3 & 1 & 4 & 5 \\ 3 & 4 & 0 & 0 \\ 3 & 2 & 0 & 0 \end{bmatrix},$$

$$F_4 = \begin{bmatrix} 5 & 1 & 0 & 3 \\ 0 & 5 & 0 & 3 \\ 0 & 4 & 0 & 0 \\ 6 & 1 & 0 & 0 \end{bmatrix}, F_5 = \begin{bmatrix} 6 & 0 & 3 & 4 \\ 6 & 2 & 4 & 2 \\ 6 & 3 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{bmatrix}, F_6 = \begin{bmatrix} 4 & 4 & 1 & 1 \\ 3 & 0 & 0 & 3 \\ 3 & 6 & 0 & 0 \\ 1 & 2 & 0 & 0 \end{bmatrix}, F_7 = \begin{bmatrix} 6 & 3 & 2 & 3 \\ 4 & 4 & 0 & 6 \\ 2 & 3 & 1 & 3 \\ 6 & 4 & 0 & 6 \end{bmatrix}$$

P_1 and P_2 represent the HFE component, $P_3 \to P_6$ represent the rainbow component, and P_7 represents the plus component. With the public key generated by $\mathcal{P} = \mathcal{T} \circ \mathcal{F} \circ \mathcal{U}$, its matrix form over $\mathbb{F}_7$ is:

$$P_1 = \begin{bmatrix} 1 & 1 & 2 & 5 \\ 1 & 2 & 3 & 2 \\ 3 & 2 & 4 & 4 \\ 3 & 3 & 0 & 3 \end{bmatrix}, P_2 = \begin{bmatrix} 0 & 2 & 0 & 6 \\ 4 & 5 & 2 & 0 \\ 6 & 3 & 3 & 4 \\ 3 & 1 & 2 & 2 \end{bmatrix}, P_3 = \begin{bmatrix} 2 & 3 & 1 & 4 \\ 4 & 5 & 4 & 5 \\ 3 & 5 & 5 & 1 \\ 5 & 1 & 0 & 6 \end{bmatrix},$$

$$P_4 = \begin{bmatrix} 0 & 6 & 0 & 2 \\ 1 & 3 & 0 & 2 \\ 5 & 1 & 5 & 1 \\ 5 & 3 & 0 & 5 \end{bmatrix}, P_5 = \begin{bmatrix} 4 & 3 & 2 & 3 \\ 6 & 5 & 2 & 4 \\ 4 & 3 & 1 & 5 \\ 5 & 2 & 4 & 5 \end{bmatrix}, P_6 = \begin{bmatrix} 1 & 4 & 2 & 2 \\ 3 & 3 & 6 & 2 \\ 5 & 4 & 0 & 0 \\ 3 & 5 & 5 & 4 \end{bmatrix}, P_7 = \begin{bmatrix} 1 & 3 & 6 & 0 \\ 0 & 3 & 4 & 0 \\ 1 & 2 & 4 & 2 \\ 2 & 1 & 6 & 4 \end{bmatrix}$$

Given the following plaintext, $(2,6,1,5)$, the resulting ciphertext is $(0,0,1,3,0,4,0)$.

Decryption: Given a ciphertext $(0,0,1,3,0,4,0)$, the following process is how you would obtain its corresponding plaintext.

Part of the secrect key:

$$\mathcal{T}^{-1} = \begin{bmatrix} 1 & 6 & 4 & 2 & 2 & 2 & 5 \\ 5 & 4 & 4 & 6 & 0 & 5 & 2 \\ 5 & 3 & 5 & 2 & 3 & 2 & 4 \\ 5 & 6 & 5 & 5 & 2 & 1 & 1 \\ 2 & 5 & 4 & 2 & 1 & 5 & 2 \\ 2 & 5 & 6 & 6 & 3 & 5 & 5 \\ 1 & 2 & 5 & 4 & 4 & 0 & 5 \end{bmatrix}, \mathcal{U}^{-1} = \begin{bmatrix} 4 & 5 & 2 & 1 \\ 3 & 1 & 3 & 1 \\ 4 & 1 & 2 & 0 \\ 5 & 6 & 1 & 1 \end{bmatrix}$$

Feed the ciphertext through $\mathcal{T}^{-1}$ to get

$$(0,6,2,6,0,4,6) \tag{5.6}$$

The first $d = 2$ elements are the corresponsing HFE outputs. Take these elements and adjust the HFE core map as follows:

$$f := f - 0\xi^{1-1} - 6\xi^{2-1} = \xi^{12}X^{14} + \xi^{6}X^{8} + \xi^{29}X^{2} + \xi$$

Perform the Berlekamp algorithm to find the preimage of f. In doing so in this toy example, you get $(0,6)$. Next, construct the vector:

$$\overline{u} = [0, 6, u_1, u_2].$$

Construct equations of the form $\overline{u}F_1\overline{u}^\top = x_i$ where x_i refers to the i^{th} element of (5.6), for $i \in \{3,4,5,6\}$. This will result with the following equations:

$$\begin{bmatrix} 6u_1 + 1 \\ 3u_1 + 3u_2 + 5 \\ 2u_2 + 2 \\ u_1 + 2u_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 6 \\ 0 \\ 4 \end{bmatrix}$$

Solving this system of equations gives us $u_1 = 6$ and $u_2 = 6$. Thus,

$$\overline{u} = [0, 6, 6, 6].$$

Finally, feed this through $\mathcal{U}^{-1}$ to get the plaintext, $[2,6,1,5]$.

REFERENCES

[1] M. Bardet, Jean-Charles Faugre, B. Salvy, and Bo-Yin Yang, *Asymptotic behaviour of the degree of regularity of semi-regular polynomial systems*, In MEGA '05, 2005. Eighth International Symposium On Effective Methods In Algebraic Geometry, 2005.

[2] J.A. Beachy and W.D. Blair, *Abstract algebra*, Waveland Press, 2006.

[3] E. R. Berlekamp, *Factoring polynomials over finite fields*, The Bell System Technical Journal **46** (1967), no. 8, 1853–1859.

[4] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret, *Cryptanalysis of hfe, multi-hfe and variants for odd and even characteristic*, Designs, Codes and Cryptography **69** (2013), no. 1, 1–52.

[5] Luk Bettale, Jean-Charles Faugere, and Ludovic Perret, *Cryptanalysis of hfe, multi-hfe and variants for odd and even characteristic*, Designs, Codes and Cryptography **69** (2013), no. 1, 1–52.

[6] Luk Bettale, Jean-Charles Faugère, and Ludovic Perret, *Cryptanalysis of HFE, multi-HFE and variants for odd and even characteristic*, Des. Codes Cryptography **69** (2013), no. 1, 1–52.

[7] Wieb Bosma, John Cannon, and Catherine Playoust, *The magma algebra system i: The user language*, J. Symb. Comput. **24** (1997), no. 3-4, 235–265.

[8] ——, *The Magma algebra system. I. The user language*, J. Symbolic Comput. **24** (1997), no. 3-4, 235–265, Computational algebra and number theory (London, 1993). MR MR1484478

[9] Bruno Buchberger, *Bruno buchbergers phd thesis 1965: An algorithm for finding the basis elements of the residue class ring of a zero dimensional polynomial ideal*, Journal of Symbolic Computation **41** (2006), 475–511.

[10] Daniel Cabarcas, Daniel Smith-Tone, and Javier A. Verbel, *Key recovery attack for ZHFE*, Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings, 2017, pp. 289–308.

[11] Ryann Cartor, Ryan Gipson, Daniel Smith-Tone, and Jeremy Vates, *On the differential security of the hfev- signature primitive*, Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings, 2016, pp. 162–181.

[12] Crystal Clough, John Baena, Jintai Ding, Bo-Yin Yang, and Ming-Shing Chen, *Square, a New Multivariate Encryption Scheme*, CT-RSA, 2009, pp. 252–264.

[13] Taylor Daniels and Daniel Smith-Tone, *Differential properties of the hfe cryptosystem*, pp. 59–75, Springer International Publishing, Cham, 2014.

[14] N. Dattani and N. Bryans, *Quantum factorization of 56153 with only 4 qubits*, Cornell University Library (2014), "https://arxiv.org/abs/1411.6758v3".

[15] W. Diffie and M. Hellman, *New directions in cryptography*, IEEE Trans. Inf. Theor. **22** (2006), no. 6, 644–654.

[16] J. Ding and D. Schmidt, *Rainbow, a new multivariable polynomial signature scheme*, ACNS 2005, LNCS **3531** (2005), 164–175.

[17] Jintai Ding and Timothy J. Hodges, *Inverting HFE systems is quasi-polynomial for all fields*, Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings, 2011, pp. 724–742.

[18] Jintai Ding and Thorsten Kleinjung, *Degree of regularity for hfe-*.

[19] Jintai Ding, Albrecht Petzoldt, and Lih-chung Wang, *The cubic simple matrix encryption scheme*, Post-Quantum Cryptography - 6th International Workshop, PQCrypto 2014, Waterloo, ON, Canada, October 1-3, 2014. Proceedings, 2014, pp. 76–87.

[20] Vivien Dubois, Pierre-Alain Fouque, Adi Shamir, and Jacques Stern, *Practical cryptanalysis of sflash*, pp. 1–12, Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[21] D.S. Dummit and R.M. Foote, *Abstract algebra*, Wiley, 2004.

[22] J. C. Faugere, *A new efficient algorithm for computing grobner bases (f4)*, Journal of Pure and Applied Algebra **139** (1999), 61–88.

[23] Jean-Charles Faugère, Mohab Safey El Din, and Pierre-Jean Spaenlehauer, *Computing loci of rank defects of linear matrices using gröbner bases and applications to cryptology*, Symbolic and Algebraic Computation, International Symposium, ISSAC 2010, Munich, Germany, July 25-28, 2010, Proceedings, 2010, pp. 257–264.

[24] Jean-Charles Faugère, Mohab Safey El Din, and Pierre-Jean Spaenlehauer, *Computing loci of rank defects of linear matrices using grÖbner bases and applications to cryptology*, Proceedings of the 2010 International Symposium on Symbolic and Algebraic Computation (New York, NY, USA), ISSAC '10, ACM, 2010, pp. 257–264.

[25] Jean-Charles Faugère and Antoine Joux, *Algebraic cryptanalysis of hidden field equation (hfe) cryptosystems using gröbner bases*, Advances in Cryptology - CRYPTO 2003 (Berlin, Heidelberg) (Dan Boneh, ed.), Springer Berlin Heidelberg, 2003, pp. 44–60.

[26] Ralf Frberg, *An inequality for hilbert series of graded algebras.*, MATHEMATICA SCANDINAVICA **56** (1985), no. 0, 117–144.

[27] Michael R. Garey and David S. Johnson, *Computers and intractability; a guide to the theory of np-completeness*, W. H. Freeman & Co., New York, NY, USA, 1990.

[28] Louis Goubin and Nicolas Courtois, *Cryptanalysis of the ttm cryptosystem*, ASIACRYPT, 2000, pp. 44–57.

[29] Lester S. Hill, *Cryptography in an algebraic alphabet*, The American Mathematical Monthly **36** (1929), 135–154.

[30] Yasuhika Ikematsu, Ray Perlner, Daniel Smith-Tone, Tsuyoshi Takagi, and Jeremy Vates, *Hferp - a new multivariate encryption scheme*, Post-Quantum Cryptography - 9th International Workshop, PQCrypto 2018, Fort Lauderdale, Florida, April 9-11, 2018, Proceedings, 2018, Accepted.

[31] Russ Juskalian, *Practical quantum computers*, MIT Technology Review:10 Breakthrough Technologies (2017).

[32] A. Kipnis, J. Patarin, and L. Goubin, *Unbalanced oil and vinegar signature schemes*, EUROCRYPT 1999. LNCS **1592** (1999), 206–222.

[33] Aviad Kipnis and Adi Shamir, *Cryptanalysis of the oil and vinegar signature scheme*, pp. 257–266, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998.

[34] ______, *Cryptanalysis of the hfe public key cryptosystem by relinearization*, Advances in Cryptology — CRYPTO' 99 (Berlin, Heidelberg) (Michael Wiener, ed.), Springer Berlin Heidelberg, 1999, pp. 19–30.

[35] N. Koblitz, A.J. Menezes, Y.H. Wu, and R.J. Zuccherato, *Algebraic aspects of cryptography*, Algorithms and Computation in Mathematics, Springer Berlin Heidelberg, 2004.

[36] Neal Koblitz, *A course in number theory and cryptography*, Springer-Verlag New York, Inc., New York, NY, USA, 1987.

[37] Tsutomu Matsumoto and Hideki Imai, *Public quadratic polynomial-tuples for efficient signature-verification and message-encryption*, pp. 419–453, Springer Berlin Heidelberg, Berlin, Heidelberg, 1988.

[38] Alfred Menezes, Ian Blake, XuHong Gao, Ronald Mullins, Scott Vanstone, and Tomik Yaghoobian, *Applications of finite fields*, Springer US, 1993.

[39] Dustin Moody, Ray A. Perlner, and Daniel Smith-Tone, *An asymptotically optimal structural attack on the ABC multivariate encryption scheme*, Post-Quantum Cryptography - 6th International Workshop, PQCrypto 2014, Waterloo, ON, Canada, October 1-3, 2014. Proceedings, 2014, pp. 180–196.

[40] ______, *Key recovery attack on the cubic ABC simple matrix multivariate encryption scheme*, Selected Areas in Cryptography - SAC 2016 - 23rd International Conference, St. John's, NL, Canada, August 10-12, 2016, Revised Selected Papers, 2016, pp. 543–558.

[41] ______, *Improved attacks for characteristic-2 parameters of the cubic ABC simple matrix encryption scheme*, Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings, 2017, pp. 255–271.

[42] ———, *Key recovery attack on the cubic abc simple matrix multivariate encryption scheme*, Selected Areas in Cryptography – SAC 2016: 23rd International Conference, Revised Selected Papers, LNCS, Springer, 2017.

[43] Crypto Museum, *History of the enigma*, 2012, "http://www.cryptomuseum.com/crypto/enigma/hist.htm".

[44] NIST, *Call for proposals*, 2017, "https://csrc.nist.gov/Projects/Post-Quantum-Cryptography/Post-Quantum-Cryptography-Standardization/Call-for-Proposals".

[45] P. C. Van Oorschot and S. A. Vanstone, *A geometric approach to root finding in gt(qm)*, IEEE Transactions on Information Theory **35** (1989), no. 2, 444–453.

[46] J. Patarin, *The oil and vinegar algorithm for signatures*, Presented at the Dagsthul Workshop on Cryptography (1997).

[47] Jacques Patarin, *Cryptanalysis of the matsumoto and imai public key scheme of eurocrypt'88*, pp. 248–261, Springer Berlin Heidelberg, Berlin, Heidelberg, 1995.

[48] ———, *Hidden fields equations (hfe) and isomorphisms of polynomials (ip): Two new families of asymmetric algorithms*, pp. 33–48, Springer Berlin Heidelberg, Berlin, Heidelberg, 1996.

[49] Jacques Patarin, *The oil and vinegar algorithm for signatures*, 1997, Presented at the Dagsthul Workshop on Cryptography.

[50] Jacques Patarin, Louis Goubin, and Nicolas Courtois, C^*_{-+} *and HM: Variations Around Two Schemes of T. Matsumoto and H. Imai*, ASIACRYPT, 1998, pp. 35–49.

[51] Ray Perlner and Daniel Smith-Tone, *A classification of differential invariants for multivariate post-quantum cryptosystems*, Post-Quantum Cryptography (Berlin, Heidelberg) (Philippe Gaborit, ed.), Springer Berlin Heidelberg, 2013, pp. 165–173.

[52] Ray A. Perlner, Albrecht Petzoldt, and Daniel Smith-Tone, *Total break of the srp encryption scheme*, Springer, In press., 2017.

[53] Albrecht Petzoldt, Ming-Shing Chen, Bo-Yin Yang, Chengdong Tao, and Jintai Ding, *Design principles for hfev- based multivariate signature schemes*, Advances in Cryptology - ASIACRYPT 2015 - 21st International Conference on the Theory and Application of Cryptology and Information Security, Auckland, New Zealand, November 29 - December 3, 2015, Proceedings, Part I, 2015, pp. 311–334.

[54] Albrecht Petzoldt, Ming-Shing Chen, Bo-Yin Yang, Chengdong Tao, and Jintai Ding, *Design principles for hfev- based multivariate signature schemes*, pp. 311–334, Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[55] Jaiberth Porras, John Baena, and Jintai Ding, *ZHFE, A new multivariate public key encryption scheme*, Post-Quantum Cryptography - 6th International Workshop, PQCrypto 2014, Waterloo, ON, Canada, October 1-3, 2014. Proceedings, 2014, pp. 229–245.

[56] Quynh H. Dang, *FIPS PUB 180-4: Secure hash standard.*

[57] R. Rivest, *The md5 message-digest algorithm*, 1992.

[58] R. L. Rivest, A. Shamir, and L. Adleman, *A method for obtaining digital signatures and public-key cryptosystems*, Commun. ACM **21** (1978), no. 2, 120–126.

[59] Ronald L. Rivest, *The md4 message digest algorithm*, Advances in Cryptology-CRYPT0' 90 (Berlin, Heidelberg) (Alfred J. Menezes and Scott A. Vanstone, eds.), Springer Berlin Heidelberg, 1991, pp. 303–311.

[60] P. Sawer, *The unsung genius who secured britain's computer defences and paved the way for safe online shopping*, The Telegraph (2016), "https://www.telegraph.co.uk/history/12191473/The-unsung-genius-who-secured-Britains-computer-defences-and-paved-the-way-for-safe-online-shopping.html".

[61] A. Shamir and A. Kipnis, *Cryptanalysis of the oil & vinegar signature scheme*, CRYPTO 1998. LNCS **1462** (1998), 257–266.

[62] P. W. Shor, *Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer*, SIAM J. Sci. Stat. Comp. **26, 1484** (1997).

[63] Daniel Smith-Tone, *Properties of the discrete differential with cryptographic applications*, Post-Quantum Cryptography (Berlin, Heidelberg) (Nicolas Sendrier, ed.), Springer Berlin Heidelberg, 2010, pp. 1–12.

[64] Chengdong Tao, Adama Diene, Shaohua Tang, and Jintai Ding, *Simple matrix scheme for encryption*, PQCrypto, 2013, pp. 231–242.

[65] Tiffany Trader, *Google chases quantum supremacy with 72-qubit processor*, 2018, "https://www.hpcwire.com/2018/03/07/google-chases-quantum-supremacy-with-72-qubit-processor/".

[66] Jeremy Vates and Daniel Smith-Tone, *Key recovery attack for all parameters of HFE-*, Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings, 2017, pp. 272–288.

[67] ______, *Key recovery attack for all parameters of HFE-*, Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings, 2017, pp. 272–288.

[68] Joachim von zur Gathen and Victor Shoup, *Computing frobenius maps and factoring polynomials*, computational complexity **2** (1992), no. 3, 187–224.

[69] Bo-Yin Yang and Jiun-Ming Chen, *Theoretical analysis of XL over small fields*, Information Security and Privacy: 9th Australasian Conference, ACISP 2004, Sydney, Australia, July 13-15, 2004. Proceedings, 2004, pp. 277–288.

[70] Takanori Yasuda and Kouichi Sakurai, *A multivariate encryption scheme with rainbow*, pp. 236–251, Springer International Publishing, Cham, 2016.

APPENDIX A - BERLEKAMP ALGORITHM

In 1967, Elwyn Berlekamp developed an efficient algorithm for factoring polynomials which ocur over finite fields, see [3]. For reference, a description of the algorithm is provided below.

The algorithm takes a square-free polynomial (a polynomial with no repeated factors), $f(x)$, as an input. This polynomial is of degree n with coefficients in $\mathbb{F}_q$. The output of the algorithm is a polynomial, $g(x)$, such that $g(x) \mid f(x)$. This process is then repeated until $f(x)$ is factored into irreducible polynomials.

Note that all factors of $f(x)$ that the algorithm searches for live within

$$R = \mathbb{F}_q[x]/\langle f(x) \rangle.$$

Specifically, the algorithm spends its time working with polynomials, $g(x)$, of the form

$$g(x)^q \equiv g(x) \pmod{f(x)}$$

which form a subalgebra of R, titled the Berlekamp subalgebra. The reason this algebra is of interest is due to the fact that the polynomials it contains satisfy the following property related to $f(x)$:

$$f(x) = \prod_{s \in \mathbb{F}_q} gcd(f(x), g(x) - s) \tag{5.7}$$

It is worth noting that every factor in the above product will not necessarily be a non-trivial of $f(x)$. Yet, there will be some, thus generating a factor of $f(x)$.

This algorithm starts by computing a basis for the Berlekamp subalgebra. This process is able to be done since it is known that the Berlekamp subalgebra is the kernel of a specific $n \times n$ matrix over $\mathbb{F}_q$. This matrix is derived from the "Berlekamp matrix" of the polynomial, represented as $\mathcal{Q} = [q_{i,j}]$, where $q_{i,j}$ is the coefficient of the jth power term in the reduction of x^{iq} modulo $f(x)$:

$$x^{iq} = q_{i,n-1}x^{n-1} + q_{i,n-2}x^{n-2} + \cdots + q_{i,0}$$

Given a special polynomial $g(x) \in R$, we can say

$$g(x) = g_{n-1}x^{n-1} + g_{n-2}x^{n-2} + \cdots + g_0,$$

which gives us the following row-vector representation of g:

$$g = \{g_0, g_1, \ldots, g_n - 1\}.$$

With the above notations, it is known that a polynomial $g(x) \in R$ is in the Berlekamp subalgebra if and only if $g(\mathcal{Q} - \mathcal{I}) = 0$. This is equivalent to stating that a polynomial $g(x) \in R$ is in the Berlekamp subalgebra if and only if it is in the null space of $\mathcal{Q} - \mathcal{I}$.

The algorithm spends its time computing the matrix $\mathcal{Q} - \mathcal{I}$, reducing it to reduced row echelon form, then finding the basis for the null space. This gives us a basis for the Berlekamp subalgebra and the ability to construct polynomials within it. Then, it computes the *gcd*'s of the form 5.7 until a non-trivial factor is found. This computation of *gcd*'s can be done using the Euclidean Algorithm since we are working in a Euclidean domain.

APPENDIX B - HASH FUNCTIONS

Within the world of cryptography, it is often necessary for a computer to quickly be able to compress data to a fixed size. This is able to be done through hash functions:

Definition 5.1. *A* ***hash function****,* $h(x) : \mathcal{A} \to \mathcal{B}$ *where* $|\mathcal{B}| \leq |\mathcal{A}|$*, which ideally has the following properties:*

- *If* $x = y$ *then* $h(x) = h(y)$ *for all* $x, y \in \mathcal{A}$*.*
- $h(x)$ *is quick to compute for all* x*.*
- *Solving* $h(x) = y$ *with knowing* y *is very difficult (preferably infeasible).*
- $h(x)$ *and* $h(x + \epsilon)$ *cannot be correlated with one another for any value* ϵ*.*
- *If* $x \neq y$*, then* $h(x) \neq h(y)$ *for all* $x, y \in \mathcal{A}$*.*

There are many different types of hash functions within cryptography that serve different purposes with different security advantages/disadvantages. I am providing details one such hash function that have been used in recent years. For recent standardizations of hash functions, see [56].

MD5

MD5 was proposed by R. Rivest in April of 1992, see [57]. This was a continuation of his work on the MD4 algorithm in [59]. This algorithm is designed very efficiently on 32-bit machines and does not require large substitution tables. It is worth noting that this hash function has been broken. However, it is a worthwhile example of hash functions. In this proposal, Rivest used the following terminology:

Definition 5.2 (see [57]). *Let X and Y represent words as defined below.*

- *A word is a 32-bit quantity and a byte is an 8-bit quantity.*
- *Let "+" denote the additions of words, modulo-*2^{32}.
- *Let $X <<< s$ denote the 32-bit value by rotating X left by s-bit positions.*
- *Let $Snot(X)$ denote the bit-wise complement of X.*
- *Let $X \vee Y$ represent the bit-wise OR of X and Y.*
- *Let X xor Y denote the bit-wise XOR of X and Y.*
- *Let XY denote the bit-wise AND of X and Y.*

The algorithm assumes that a b-bit message is the input, and the goal is to compute its hash function value, a 128-bit "fingerprint". Here, the algorithm allows b to be any nonnegative integer and the input of the b-bit message in the following form: $m_0m_1 \dots m_{b-1}$. What follows is the step by step process in which the algorithm processes the input and produces the desired 128-bit output.

Step 1: Append Padding Bits

The input is padded in such a way that the result is congruent to 448 modulo 512. This results in the padded message being 64-bits less than being a multiple of 512. This padding is esential, reguardless of initial length of message (including if the message is originally congruent to 448 modulo 512).The padded message is done to result in the following:

$$m_0m_1 \dots m_{b-1}10 \dots 0$$

where the number of 0's are added so that the padded message is the desired 448 modulo 512. This can require from a 1-bit addition to, at most, 512-bits being appended.

Step 2: Append Length

Next, a 64-bit string is appended to the padded result where said string is a 64-bit representation of b. If $b > 2^{64}$, then only use the lower 64-bits of the base 2 representation of b are used. This results in our current value being a multiple of 512, which also implies it has a length that is a multiple of 16 (32-bit) words. Let N represent said multiple of 16 and $M[0 \ldots N-1]$ represent the collection of 32-bit words.

Step 3: Initialize MD Buffer

Next, a 4-word buffer is used to compute the message digest. These are initialized in the following hexadecimal, low-order bytes:

$$A : 01\ 23\ 45\ 67$$

$$B : 89\ ab\ cd\ ef$$

$$C : fe\ dc\ ba\ 98$$

$$D : 76\ 54\ 32\ 10$$

Step 4: Process Message in 16-Word Blocks]

Next, define the following functions:

$$F(X,Y,Z) = XY \vee not(X)Z$$

$$G(X,Y,Z) = XZ \vee Y not(Z)$$

$$H(X,Y,Z) = X\ xor\ Y\ xor\ Z$$

$$I(X,Y,Z) = Y\ xor\ (X \vee not(Z))$$

Then, the algorithm is performed as follows. This is the exact same code as provided in [57] and it uses a 64-element table constructed from the sine function. If one wishes to practice using this hash function, refer to [57] for those values.

```
/* Process each 16-word block. */
For i = 0 to N/16-1 do
/* Copy block i into X. */
For j = 0 to 15 do
Set X[j] to M[i*16+j].
end /* of loop on j */
/* Save A as AA, B as BB, C as CC, and D as DD. */
AA = A
BB = B
Rivest [Page 4]
RFC 1321 MD5 Message-Digest Algorithm April 1992
CC = C
DD = D
/* Round 1. */
/* Let [abcd k s i] denote the operation
a = b + ((a + F(b,c,d) + X[k] + T[i]) <<< s). */
/* Do the following 16 operations. */
[ABCD 0 7 1] [DABC 1 12 2] [CDAB 2 17 3] [BCDA 3 22 4]
[ABCD 4 7 5] [DABC 5 12 6] [CDAB 6 17 7] [BCDA 7 22 8]
[ABCD 8 7 9] [DABC 9 12 10] [CDAB 10 17 11] [BCDA 11 22 12]
[ABCD 12 7 13] [DABC 13 12 14] [CDAB 14 17 15] [BCDA 15 22 16]
/* Round 2. */
/* Let [abcd k s i] denote the operation
a = b + ((a + G(b,c,d) + X[k] + T[i]) <<< s). */
/* Do the following 16 operations. */
[ABCD 1 5 17] [DABC 6 9 18] [CDAB 11 14 19] [BCDA 0 20 20]
[ABCD 5 5 21] [DABC 10 9 22] [CDAB 15 14 23] [BCDA 4 20 24]
```

```
[ABCD 9 5 25] [DABC 14 9 26] [CDAB 3 14 27] [BCDA 8 20 28]
[ABCD 13 5 29] [DABC 2 9 30] [CDAB 7 14 31] [BCDA 12 20 32]
/* Round 3. */
/* Let [abcd k s t] denote the operation
a = b + ((a + H(b,c,d) + X[k] + T[i]) <<< s). */
/* Do the following 16 operations. */
[ABCD 5 4 33] [DABC 8 11 34] [CDAB 11 16 35] [BCDA 14 23 36]
[ABCD 1 4 37] [DABC 4 11 38] [CDAB 7 16 39] [BCDA 10 23 40]
[ABCD 13 4 41] [DABC 0 11 42] [CDAB 3 16 43] [BCDA 6 23 44]
[ABCD 9 4 45] [DABC 12 11 46] [CDAB 15 16 47] [BCDA 2 23 48]
/* Round 4. */
/* Let [abcd k s t] denote the operation
a = b + ((a + I(b,c,d) + X[k] + T[i]) <<< s). */
/* Do the following 16 operations. */
[ABCD 0 6 49] [DABC 7 10 50] [CDAB 14 15 51] [BCDA 5 21 52]
[ABCD 12 6 53] [DABC 3 10 54] [CDAB 10 15 55] [BCDA 1 21 56]
[ABCD 8 6 57] [DABC 15 10 58] [CDAB 6 15 59] [BCDA 13 21 60]
[ABCD 4 6 61] [DABC 11 10 62] [CDAB 2 15 63] [BCDA 9 21 64]
/* Then perform the following additions. (That is increment each
of the four registers by the value it had before this block
was started.) */
A = A + AA
B = B + BB
C = C + CC
D = D + DD
end /* of loop on i */
```

APPENDIX C - PSEUDOCODE

```
Buchberger Algorithm
Input: $G \subset \mathbb{F}[X] = \mathbb{F}[X_1, \dots, X_n]$
Output: $GB$, a Gröbner Basis for the ideal $I = \langle G \rangle$ in $\mathbb{F}[X]$
GB:={};
while $GB \neq G$
    $GB := G$.
    for every $g_i, g_j \in G$ where $i \neq j$
        Compute $S_{i,j} = S(g_i, g_j)$.
        Compute $S_{i,j} \xrightarrow{G} S'_{i,j}$.
        if $S'_{i,j} \neq 0$, then Append $S'_{i,j}$ to $G$.
    end for
```

F4: Symbolic Preprocessing

```
Input: $L \subset T \times R[X]$ and $G \subset R[X]$
Output: a finite subset of $R[X]$.
F:=$\{t * f \mid (t, f) \in L\}$
Done:=HT(F)
while $T(F) \neq Done$ do
  $m$ an element of $T(F) \setminus Done$
  $Done := Done \cup \{m\}$
  if $m$ top reducible module $G$ then
    $m = m' * HT(f)$ for some $f \in G$ and some $m' \in T$
    $F := F \cup \{m' * f\}$
return $F$
```

F4: Reduction

```
Input:$L \subset T \times R[X]$ and $G \subset R[X]$
Output: a finite subset of $R[X]$, possibly empty
$F :=$ SymbolicPreprocessing(L,G)
$\tilde{F} :=$ Reduction to Row Echelon Form of $F$ w.r.t. $<$
$\tilde{F}^{+} := \{f \in \tilde{F} \mid HT(f) \notin HT(F)\}$
```

F4: Algorithm

Input: $F \subset R[X]$ and Sel, a chosen selection function from [22]

Output: a finite subset of $R[X]$

```
$G := F$, $\tilde{F}_0^+ := F$, $d := 0$
$P := \{Pair(f,g) \mid f,g \in G \text{ with } f \neq g\}$
while $P \neq \varnothing$ do
    $d := d + 1$
    $P_d := Sel(P)$
    $P := P \smallsetminus P_d$
    $L - d := Left(P_d) \cup Right(P_d)$
    $\tilde{F}_d^+ := Reduction(L_d, G)$
    for $h \in \tilde{F}_d^+$ do
        $P := P \cup \{Pair(h,g) \mid g \in G\}$
        $G := G \cup \{h\}$
return $G$
```

HFEvKeyCheck

Input: An $HFEv$ central map f, a flag flg

Output: Set of indices of coefficients m_i of submatrix m_{00} which are possibly nonzero in a linear map inducing differential symmetry for f.

```
for monomial $\alpha_{i,j}x^{q^i+q^j}$ in $f$
  $S_i = \{\}$;
  $S_j = \{\}$;
  for monomial with powers $r$ and $s$ in $f$
    $S_i = S_i \cup \{r-j, s-j, i-j+r-s, i-j+s-r\}$;
    $S_j = S_j \cup \{r-i, s-i, j-i+r-s, j-i+s-r\}$;
  end for;
end for;
if flg
then
  return all $S_i$;
else
  return $\bigcap S_i$;
end if;
```

HFEv-KeyCheck

Input: An $HFEv^{-}$ *central map* $\pi(f)$*, the corank of* π*,* r

Output: Set of indices of coefficients m_i *of submatrix* m_{00} *which are possibly nonzero in a linear map inducing differential symmetry for* $\pi(f)$*.*

```
Call: HFEvKeyCheck(f,1);
for all $S_i$
  $T_i = \{\}$;
  for $j$ from 0 to $r-1$
    $T_i = T_i \cup (j + S_i)$;
  end for;
end for;
return $\bigcap T_i$;
```

CURRICULUM VITAE

Jeremy Vates

Academic Record

University of Louisville, Louisville, KY
Ph.D. in Applied and Industrial Mathematics **Expected August 2018**
Areas of Concentration: Multivariate Public Key Cryptography
Advisor: Dr. Daniel Smith-Tone

University of Louisville, Louisville, KY
M.A. Mathematics **December 2015**

Marian University, Indianapolis, IN
B.A. Mathematics **May 2013**

Publications

Y. Ikematsu, R. Perlner, D. Smith-Tone, T. Takagi, **J. Vates**. *HFERP- A New Multivarite Encryption Scheme*, PQCRYPTO 2018, Fort Lauderdale, Florida, April 9-11, 2018, Proceedings, pages 396-416, 2018.

J.Vates & D. Smith-Tone. *Key Recovery Attack for All Parameters of HFE-*, PQCRYPTO 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings, pages 272-288, 2017.

R. Cartor, R. Gipson, D. Smith-Tone, & **J. Vates**. *On the Differential Security of the HFEv- Signature Primitive*, PQCRYPTO 2016, LNCS, Vol. 9606, Springer (2016).

Teaching Experience

- **Instructor for Johns Hopkins CTY:**
  Cryptology Summer 2017
  A three week, intense summer course for 8th and 9th graders. I developed a syllabus and covered all material with the assistance of a TA. Individual student evaluations were given at the end of the session.

- **Instructor for University of Louisville:**
  College Algebra/Precalculus/Business Calculus/
  Mathematics for Elementary Education 2014-Current
  I have been an instructor for the courses listed above, some with repetitions. I developed course structure and a syllabus for each class as well as administering final grades.

- **Teaching Assistant for University of Louisville:**
  College Algebra/Contemporary Math/
  Business Calculus/Finite Mathematics 2013-Current
  In these courses, I have taught recitations for large lectures. I have taught all of these recitations multiple times. See my teaching statement for specifics. It was common to craft quizzes and grade exams created by main instructor.

Presentations, Invited Talks, Seminars, and Service

- **PQC Introduction** *Marian University, 2015*
- **Differential Security of the HFEv-** *University of Cincinnati, 2015*
- **PQCRYPTO 2016** *Fukuoka, Japan, Spring 2016*
- **Introduction to Mathematical Proofs** *Louisville Math Circle, 2016*
- **Introduction to Cryptology** *Louisville Math Circle, 2016*

- **Referee PQCRYPTO 2017** *2017*
- **Referee PQCRYPTO 2018** *2017*

Technical Skills and Other Abilities

- **Mathematics Software:** Magma, Mathematica
- **Programming Languages:** C#, Basic, Python
- **Document Processing:** LaTex, MS Word
- **Database Software:** MS Excel, MS Access
- **Operating Systems:** Windows, Linux