University of Zurich
Institute of Mathematics

Master Thesis

A Code-Based Cryptosystem using GRS codes

Author
Violetta Weger

Supervisor
Prof. Dr. Joachim Rosenthal

Co-supervisor
Reto Schnyder

HS 2016

Contents

1 Introduction

In a public-key cryptosystem (PKC) we consider two people who want to exchange a secret key, we call the constructor Bob and the second person Alice. Bob constructs a private key and public key, which he publishes. Alice who wants to send a message to Bob, uses the public key to encrypt her message and sends the cipher to Bob. Bob can decrypt the encrypted message with the private key. Eve, the eavesdropper, only sees the public key and the encrypted message. For the cryptosystem to be secure, it should be unfeasible for Eve to reconstruct the message.

Example 1 (RSA)**.** As an example for a PKC we want to look at RSA [33]. Bob takes two distinct prime numbers p and q, computes their product $n = pq$ and the Euler-totient function of the product $\phi(n) = (p-1)(q-1)$. He chooses e a natural number smaller than $\phi(n)$, which is coprime to $\phi(n)$. Bob publishes (n, e) and keeps private (p, q). Alice encrypts her message m by computing $c = m^e \bmod n$. Bob can decrypt c by first computing d and b s.t.

$$de + b\phi(n) = 1$$

and since

$$c^d = (m^e)^d = m^{1-b\phi(n)} = m(m^{\phi(n)})^{-b} = m1^{-b} = m,$$

he can recover the message m. Eve sees n but there is no feasible algorithm to compute p and q.

Recently, the National Security Agency (NSA) and the National Institute of Standards and Technology (NIST) have announced [1, 2, 3] that it might be possible that a quantum computer will be available in 2030 capable of breaking a 1024 bits RSA key. As it is well known most public key cryptosystems used in practice rely on the hardness of factoring integers or on the discrete logarithm problem in a finite field or an elliptic curve. If a capable quantum computer could be built then using Shor's Algorithm [36] and its extensions all above used systems would become insecure.

Hence there is a huge urge for research in so called post-quantum cryptography. The most promising post-quantum candidates are lattice-based, code-based or based on multivariate quadratic equations, since they are known to rely on NP-hard problems. The book by Bernstein, Buchmann and Dahmen [7] gives a good overview to the area of post-quantum cryptography. Other good background sources are a recent survey on lattice based cryptography by Peikert [29] and a monograph by Ding, Gower and Schmidt [15] on multivariate cryptography.

Code-based cryptography first came up in the 70's by the work of McEliece. A short description of the McEliece system [26] is provided, a full description of the system can be found in Chapter 3.

The McEliece system in its original version uses a binary irreducible Goppa code Γ, which is represented by a generator matrix G and can correct up to t errors. Instead of publishing the generator matrix directly, one publishes a scrambled matrix $G' = SGP$, where S is an invertible matrix and P is a permutation matrix. The scrambling matrices and the original matrix are only known to the constructor. One can encrypt the message m, by multiplying it with the scrambled matrix G' and adding an error vector e of weight less than or equal to t, i.e. the cipher text is computed by $c = mG' + e$. The constructor can take away the permutation matrix, without changing the weight of the error vector, thus he can use the decoding algorithm of the code and recovers the message m.

The Niederreiter system [28] has a similar protocol using the parity check matrix instead of the generator matrix and syndrome decoding for decryption. The Niederreiter system and the McEliece system have equivalent security.

Niederreiter suggested to use generalized Reed-Solomon (GRS) codes, but the Niederreiter system (and therefore also the McEliece system) is broken when using GRS codes by the attack of Sidelnikov and Shestakov [37]. The McEliece cryptosystem in its original version using Goppa codes is still unbroken, but has the main drawback of having large key sizes.

As a solution towards this problem, one could take another family of codes instead, like the GRS codes or other algebraic geometric (AG) codes. There have been many attempts to use these families of codes in variants of the McEliece cryptosystem, but their algebraic structure not only reduces the key size, it may also help an attacker to reveal information of the secret code.

Baldi, Bianchi, Chiaraluce, Rosenthal and Schipani proposed in [4] a variant of the McEliece cryptosystem, in order to reconsider the use of GRS codes. The main idea of the proposal is to use, instead of a permutation matrix, the sum $T + R$, where T is a sparse matrix of row weight m and R is a matrix of rank z. This alternative scrambling is meant to hide the algebraic structure of the secret code. This thwarts the attack of Sidelnikov and Shestakov. Nevertheless Couvreur, Gaborit, Gauthier-Umaña, Otmani and Tillich presented in [13, 18, 14] for some parameters a distinguisher attack on this cryptosystem.

In this master thesis we present a variant of the McEliece system, proposed by Bolkema, Gluesing-Luerssen, Kelley, Lauter, Malmskog and Rosenthal in [9] using GRS codes as secret code and a row weight two matrix instead of a permutation matrix. Thus this variant is a special case of the proposal in [4], by setting $m = 2$ and $z = 0$. The main part of this thesis relies on showing security of this proposal against the distinguisher attacks in [13, 18, 14].

This thesis is ordered in the following way: The first part consists of a survey on code-based cryptosystems and their attacks. In the second part we analyze the new proposal [9], this consists of an argument for the security and a comparison of the key sizes. The chapters are structured as follows. In the first chapter, the introduction, we explain the motivation to the new proposal. In the second chapter we state the preliminaries, which are needed for the understanding, such as definitions of coding theory and some important properties

of codes. In the third chapter we want to look at some selected variants of the McEliece system and at their attacks. In the fourth chapter we present the new proposal. The fifth chapter is the main part, in which we analyze the security of the proposal by showing that none of the considered attacks apply. We will also provide some experimental results. In chapter six we consider possible vulnerabilities of the proposal. In chapter seven we analyze other properties of the cryptosystem such as its complexity and the key size. In the last chapter we conclude the results and state possible improvements that could be considered for future work.

2 Preliminaries

We will recall the definitions and properties of codes we need for understanding, such as Goppa codes, GRS codes, dual codes and square codes. We will start with basic definitions of coding theory.

Definition 1. [e.g. [22], Chapter 1, page 3] An $[n, k]$-linear block code over a finite field $\mathbb{F}_q$ is a k-dimensional linear subspace $C \subseteq \mathbb{F}_q^n$. There exists a $k \times n$ generator matrix G and a $(n-k) \times n$ parity check matrix H defined by the properties:

$$C = \left\{ uG \;\middle|\; u \in \mathbb{F}_q^k \right\} = \left\{ x \in \mathbb{F}_q^n \;\middle|\; Hx^T = \mathbf{0} \right\}.$$

Where $\mathbf{0}$ denotes the zero vector. We will sometimes write $\langle G \rangle = C$.

Definition 2. [e.g. [22], Chapter 1, page 8] Let $x, y \in \mathbb{F}_q^n$. We define their Hamming distance to be

$$d_H(x, y) = \left| \left\{ i \in \{1, \ldots, n\} \;\middle|\; x_i \neq y_i \right\} \right|.$$

Definition 3. [e.g. [22], Chapter 1, page 8] Let $x \in \mathbb{F}_q^n$. The weight of x is defined as

$$\mathrm{wt}(x) = \left| \left\{ i \in \{1, \ldots, n\} \;\middle|\; x_i \neq 0 \right\} \right|.$$

Observe that $\mathrm{wt}(x) = d_H(x, \mathbf{0})$.

Let in the following C be an $[n, k]$-linear block code over $\mathbb{F}_q$.

Definition 4. [e.g. [22], Chapter 1, page 9] We define the minimum distance of C to be

$$d(C) = \min \left\{ d_H(x, y) \;\middle|\; x, y \in C, x \neq y \right\}.$$

Which is by above observation equivalent to

$$d(C) = \min \left\{ \mathrm{wt}(x) \;\middle|\; x \in C, x \neq \mathbf{0} \right\}.$$

Definition 5. [e.g. [22], Chapter 1, page 26] We denote by $C^\perp$ the dual code of C, defined as

$$C^\perp = \left\{ x \in \mathbb{F}_q^n \;\middle|\; x \cdot y = 0 \ \forall y \in C \right\}.$$

Where throughout this thesis $x \cdot y$ denotes the scalar product

$$x \cdot y = \sum_{i=1}^{n} x_i y_i.$$

Observe that the generator matrix of $C^\perp$ is the parity check matrix of C. $C^\perp$ is therefore an $[n, n-k]$ linear code over $\mathbb{F}_q$.

Lemma 1. [e.g. [22], Chapter 1, Theorem 10] Let H be a parity check matrix of C of size $(n-k) \times n$. Then C has minimum distance d, if, and only if every $d-1$ columns of H are linearly independent and some d columns are linearly dependent.

We provide a proof of this lemma, which can also be found in [22], Chapter 1, Theorem 10.

Proof. Recall that the minimum distance of C is given by the minimal weight of the nonzero codewords, i.e.

$$d(C) = \min\left\{\mathrm{wt}(c) \;\middle|\; c \in C,\ c \neq \mathbf{0}\right\}.$$

Let c be a nonzero codeword of C, then by definition $Hc^T = \mathbf{0}$. Then c has weight w if, and only if w columns of H are linearly dependent. □

With this few definitions we can already state the first theorem, concerning a bound for the minimum distance of a code, the Singleton bound.

Theorem 1 (Singleton Bound)**.** [e.g. [22], Chapter 1, Theorem 11] Let C be an $[n, k]$-linear block code. Then $d(C) \leq n - k + 1$.

For the proof of the Singleton bound we follow the idea of [22], Chapter 1, Theorem 11.

Proof. Let H be the $(n - k) \times n$ parity check matrix of C. By Lemma 1 we have that the distance of a code is given by the minimal number d, s.t. there exist d columns of H which are linearly dependent. Since H has rank $n - k$, any $n - k + 1$ columns are linearly dependent and we get the claim. □

There exist codes, which reach this bound and they are of a great interest for coding theory.

Definition 6. [e.g. [22], Chapter 1, page 33] A $[n, k]$-linear block code C, with $d(C) = n - k + 1$ is called a maximum distance separable (MDS) code.

Codes have an error correction capacity, which depends on their minimum distance.

Theorem 2. [e.g. [22], Chapter 1, Theorem 2] Let C be an $[n, k]$-linear block code with minimum distance d. Then C can correct up to $\lfloor \frac{d-1}{2} \rfloor$ errors.

For the proof of this theorem we follow the idea of [22], Chapter 1, Theorem 2.

Proof. For $c \in C$ define the ball of radius r and center c to be

$$B_r(c) = \left\{c' \in C \;\middle|\; d_H(c, c') \leq r\right\}.$$

Let $t = \lfloor \frac{d-1}{2} \rfloor$. Since the minimum distance of the code is d, we know that balls with radius t around a codeword do not overlap. Hence if we received x and $d_H(x, C) \leq t$, then there exists a unique codeword $c \in C$, s.t. $d_H(x, c) \leq t$. Since if there would exist another $c' \in C$, which has $d_H(x, c') \leq t$, then

$$d_H(c, c') \leq d_H(c, x) + d_H(c', x) \leq 2t.$$

And since the minimum distance of two distinct codewords is $d > 2t$ we conclude $c = c'$. □

With this basic definitions we are ready to define GRS and Goppa codes and state some of their properties.

Definition 7 (Reed-Solomon Code)**.** [e.g. [22], Chapter 10, page 294] Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ integers. Let $\alpha \in \mathbb{F}_q^n$ be an n-tuple of distinct elements, i.e. $\alpha = (\alpha_1, \ldots, \alpha_n)$ with $\alpha_i \neq \alpha_j \ \forall i \neq j \in \{1, \ldots, n\}$. The Reed-Solomon code $\mathrm{RS}_{n,k}(\alpha)$ has dimension k and is the set of $(p(\alpha_1), \ldots, p(\alpha_n))$, where p ranges over all polynomials of degree less than k, having coefficients in $\mathbb{F}_q$. Thus

$$\mathrm{RS}_{n,k}(\alpha) = \left\{ (p(\alpha_1), \ldots, p(\alpha_n)) \;\middle|\; p \in \mathbb{F}_q[x],\ \deg(p) < k \right\}.$$

We can write the canonical generator matrix of $\mathrm{RS}_{n,k}(\alpha)$ as

$$G = \begin{pmatrix} 1 & \cdots & 1 \\ \alpha_1 & \cdots & \alpha_n \\ \vdots & & \vdots \\ \alpha_1^{k-1} & \cdots & \alpha_n^{k-1} \end{pmatrix}.$$

Definition 8 (Generalized Reed-Solomon Code)**.** [e.g. [22], Chapter 10, page 303] Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ integers. Let $\alpha \in \mathbb{F}_q^n$ be an n-tuple of distinct elements, i.e. $\alpha = (\alpha_1, \ldots, \alpha_n)$ with $\alpha_i \neq \alpha_j \ \forall i \neq j \in \{1, \ldots, n\}$. Let $\beta \in \mathbb{F}_q^n$ be an n-tuple of nonzero elements, i.e. $\beta = (\beta_1, \ldots, \beta_n)$, with $\beta_i \neq 0 \ \forall i \in \{1, \ldots, n\}$. The Generalized Reed-Solomon code $\mathrm{GRS}_{n,k}(\alpha, \beta)$ has dimension k and is the set of $(\beta_1 p(\alpha_1), \ldots, \beta_n p(\alpha_n))$, where p ranges over all polynomials of degree less than k, having coefficients in $\mathbb{F}_q$. Thus

$$\mathrm{GRS}_{n,k}(\alpha, \beta) = \left\{ (\beta_1 p(\alpha_1), \ldots, \beta_n p(\alpha_n)) \;\middle|\; p \in \mathbb{F}_q[x],\ \deg(p) < k \right\}.$$

We can write the canonical generator matrix of $\mathrm{GRS}_{n,k}(\alpha, \beta)$ as

$$G = \begin{pmatrix} \beta_1 & \cdots & \beta_n \\ \beta_1\alpha_1 & \cdots & \beta_n\alpha_n \\ \vdots & & \vdots \\ \beta_1\alpha_1^{k-1} & \cdots & \beta_n\alpha_n^{k-1} \end{pmatrix}.$$

We observe that GRS codes differ from RS codes only in the weight β. GRS codes build an interesting family of codes, since they reach the singleton bound and have an efficient decoding algorithm.

Let $\mathrm{GRS}_{n,k}(\alpha, \beta)$ be a GRS code as in Definition 8.

Proposition 1. [e.g. [22], Chapter 10, page 304] GRS codes are MDS codes, i.e. $d(\mathrm{GRS}_{n,k}(\alpha, \beta)) = n - k + 1$.

For the proof of this proposition we follow the idea as found in [22], Chapter 10, page 304.

Proof. The distance of a code is equal to the minimum weight of the nonzero codewords. Let c be a nonzero codeword of $\mathrm{GRS}_{n,k}(\alpha, \beta)$. We define $p(x) \in \mathbb{F}_q[x]$ to be the polynomial corresponding to c, which means that

$$c = (\beta_1 p(\alpha_1), \ldots, \beta_n p(\alpha_n)).$$

By the definition of the GRS code, we have that the degree of p is strictly less than k. Since β is an n-tuple of nonzero elements, a coordinate c_i of c can only be zero, if $p(\alpha_i) = 0$. Since p has at most $k-1$ roots, we have that the weight of c is at least $n-(k-1)$. With the Singleton bound, we get the claim. □

GRS codes have an efficient decoding algorithm, if the number of the errors t satisfies the following

$$t \leq \lfloor \frac{d(\mathrm{GRS}_{n,k}) - 1}{2} \rfloor = \lfloor \frac{n-k}{2} \rfloor.$$

For example in [39], Chapter 6.7 the decoding algorithm is explained.

Proposition 2. [e.g. [22], Chapter 10, Theorem 4] The dual code of a GRS code is again a GRS code, since

$$\mathrm{GRS}_{n,k}(\alpha, \beta)^{\perp} = \mathrm{GRS}_{n,n-k}(\alpha, \gamma).$$

Where

$$\gamma_i = \beta_i^{-1} \prod_{\substack{j=1 \\ j \neq i}}^{n} (\alpha_i - \alpha_j)^{-1}.$$

For the proof of this proposition we follow the idea of [20], Proposition 1 using Lagrange interpolation.

Proof. Given an n-tuple c we want to reconstruct the unique polynomial p associated to c, i.e.

$$c = (\beta_1 p(\alpha_1), \ldots, \beta_n p(\alpha_n)).$$

Define

$$\begin{aligned} L(x) &= \prod_{i=1}^{n} (x - \alpha_i), \\ L_i(x) &= \prod_{j \neq i} (x - \alpha_j). \end{aligned}$$

Then by Lagrange interpolation we have that

$$p(x) = \sum_{i=1}^{n} \frac{L_i(x)}{L_i(\alpha_i)} p(\alpha_i).$$

By the definition, we can write γ_i as

$$\gamma_i = \beta_i^{-1} L_i(\alpha_i)^{-1}.$$

We want to show that $c \cdot c' = 0$ for all codewords c of $\mathrm{GRS}_{n,k}(\alpha, \beta)$ and all codewords c' of $\mathrm{GRS}_{n,n-k}(\alpha, \gamma)$. Let p be the polynomial associated to c and p' the polynomial associated to c'. Therefore p is of degree strictly less than k and p' is of degree strictly less than $n-k$, hence their product pp' is of degree strictly less than $n-1$. By Lagrange interpolation we can write

$$p(x)p'(x) = \sum_{i=1}^{n} \frac{L_i(x)}{L_i(\alpha_i)} p(\alpha_i) p'(\alpha_i).$$

If we compare the coefficient of x^{n-1} we get

$$\begin{aligned} 0 &= \sum_{i=1}^{n} \frac{1}{L_i(\alpha_i)} p(\alpha_i) p'(\alpha_i) \\ &= \sum_{i=1}^{n} (\beta_i p(\alpha_i)) \left(\frac{\beta_i^{-1}}{L_i(\alpha_i)} p'(\alpha_i) \right) \\ &= \sum_{i=1}^{n} (\beta_i p(\alpha_i))(\gamma_i p'(\alpha_i)) = c \cdot c'. \end{aligned}$$

□

We have the following useful proposition.

Proposition 3. [e.g. [22], Chapter 10, page 305]

$$\mathrm{GRS}_{n,k}(\alpha, \beta) = \mathrm{GRS}_{n,k}(a\alpha + \mathbf{b}, c\beta)$$

for any $a, c \in \mathbb{F}_q^{\times}$ and $\mathbf{b} = (b, \ldots, b) \in \mathbb{F}_q^n$.

This allows us to fix for example α_1 and α_2 to be arbitrary distinct elements of $\mathbb{F}_q$.

Proof. Observe that for any $a, c \in \mathbb{F}_q^{\times}$ and $\mathbf{b} = (b, \ldots, b) \in \mathbb{F}_q^n$

$$\begin{aligned} \phi : \mathbb{F}_q &\to \mathbb{F}_q \\ x &\mapsto ax + \mathbf{b} \end{aligned}$$

and

$$\begin{aligned} \psi : \mathbb{F}_q &\to \mathbb{F}_q \\ x &\mapsto cx \end{aligned}$$

are both linear transformations. For the part

$$\mathrm{GRS}_{n,k}(\alpha, \beta) = \mathrm{GRS}_{n,k}(\alpha, c\beta)$$

observe that the generator matrix of the left hand side can be denoted by G and the generator matrix of the right hand side is then SG, where S is the invertible matrix

$$\begin{bmatrix} c & & 0 \\ & \ddots & \\ 0 & & c \end{bmatrix}.$$

Hence the code generated by G and the code generated by SG are the same. For the second part a codeword of $\mathrm{GRS}_{n,k}(\alpha, \beta)$ has the form

$$g = (\beta_1 p(\alpha_1), \ldots, \beta_n p(\alpha_n)),$$

where p is a polynomial of degree less than k. If we take $p' = p \circ \phi$, which is still of degree less than k, then a codeword of $\mathrm{GRS}_{n,k}(a\alpha + \mathbf{b}, \beta)$ is of the form

$$h = (\beta_1 p'(\alpha_1), \ldots, \beta_n p'(\alpha_n)).$$

And by the Definition 8 of GRS codes, we observe that they are the same code. □

For the Goppa code we will only state its definition and some properties like its dimension and minimum distance, since we will only consider Goppa codes in the original McEliece version.

Let m be a positive integer, $n = q^m$ and $\mathbb{F}_{q^m}$ be a finite field. Let $G \in \mathbb{F}_{q^m}[x]$. Then define the quotient ring

$$S_m = \mathbb{F}_{q^m}[x] \Big/ \langle G \rangle .$$

Lemma 2. [e.g. [22], Chapter 12, page 339] Let $\alpha \in \mathbb{F}_q$ be such that $G(\alpha) \neq 0$. Then $(x - \alpha)$ is invertible in S_m and

$$(x - \alpha)^{-1} = -\frac{1}{G(\alpha)} \frac{G(x) - G(\alpha)}{x - \alpha}.$$

For the proof of this lemma we will follow the idea of [22], Chapter 12, page 339.

Proof. By Euclid we have that

$$G(x) = f(x)(x - \alpha) + G(\alpha).$$

We observe that

$$f(x) = \frac{G(x) - G(\alpha)}{x - \alpha}.$$

Therefore in S_m we have that

$$f(x)(x - \alpha) \equiv -G(\alpha) \bmod G(x).$$

With this we get the claim, since

$$(-G(\alpha))^{-1} f(x)(x - \alpha) \equiv 1 \bmod G(x).$$

Which implies

$$(x - \alpha)^{-1} \equiv -\frac{1}{G(\alpha)} \frac{G(x) - G(\alpha)}{x - \alpha} \bmod G(x).$$

For the uniqueness of the inverse, assume that f and f' are inverses of $(x - \alpha)$, i.e.

$$(x - \alpha)f \equiv (x - \alpha)f' \equiv 1 \bmod G(x).$$

Then it follows that

$$(x - \alpha)(f - f') \equiv 0 \bmod G(x).$$

And since by Euclid $\deg(f), \deg(f') < \deg(G)$, it follows that

$$f \equiv f' \bmod G(x).$$

□

Definition 9 (Classical Goppa Code)**.** [e.g. [22], Chapter 12, page 338] Let $L = \{\alpha_1, \ldots, \alpha_n\} \subseteq \mathbb{F}_{q^m}^n$, be s.t. $\alpha_i \neq \alpha_j \ \forall i \neq j \in \{1, \ldots, n\}$ and $G(\alpha_i) \neq 0 \ \forall i \in \{1, \ldots, n\}$. Then we can define the classical q-ary Goppa code as

$$\Gamma(L, G) = \left\{ a \in \mathbb{F}_q^n \;\middle|\; \sum_{i=1}^{n} \frac{a_i}{x - \alpha_i} = 0 \text{ in } S_m \right\}.$$

Let $\Gamma(L, G)$ be a classical q-ary Goppa code as in Definition 9.

Proposition 4. [e.g. [22], Chapter 12, page 339] The Goppa code has distance $d(\Gamma(L, G)) \geq \deg(G) + 1$ and dimension $k \geq n - m\deg(G)$.

For the proof of this proposition we will follow the idea of [22], Chapter 12, page 339.

Proof. Note that

$$\deg\left(\frac{G(x) - G(\alpha_i)}{x - \alpha_i}\right) = \deg(G) - 1.$$

c is a codeword of $\Gamma(L, G)$, if, and only if

$$\sum_{i=1}^{n} c_i \frac{G(x) - G(\alpha_i)}{x - \alpha_i} \left(\frac{1}{-G(\alpha_i)}\right) = 0.$$

To get the parity check matrix of $\Gamma(L, G)$, we define $\deg(G) = r$ and write G as

$$G(x) = g_0 + \ldots + g_r x^r.$$

Observe that

$$\frac{G(x) - G(\alpha_i)}{x - \alpha_i} = g_r(x^r + x^{r-1}\alpha_i + \cdots + \alpha_i^{r-1}) + \cdots + g_2(x + \alpha_i) + g_1.$$

So we can write the parity check matrix H as

$$H = \begin{bmatrix} g_r G(\alpha_1)^{-1} & \cdots & g_r G(\alpha_n)^{-1} \\ (g_{r-1} + g_r\alpha_1)G(\alpha_1)^{-1} & \cdots & (g_{r-1} + g_r\alpha_n)G(\alpha_n)^{-1} \\ \vdots & & \vdots \\ (g_1 + \cdots + g_r\alpha_1^{r-1})G(\alpha_1)^{-1} & \cdots & (g_1 + \cdots + g_r\alpha_n^{r-1})G(\alpha_n)^{-1} \end{bmatrix}.$$

Hence $H = WXY$, where W, X, Y are the following matrices.

$$W = \begin{bmatrix} g_r & 0 & \cdots & 0 \\ g_{r-1} & g_r & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ g_1 & g_2 & \cdots & g_r \end{bmatrix}, \; X = \begin{bmatrix} 1 & \cdots & 1 \\ \alpha_1 & \cdots & \alpha_n \\ \vdots & & \vdots \\ \alpha_1^{r-1} & \cdots & \alpha_n^{r-1} \end{bmatrix}, \; Y = \begin{bmatrix} G(\alpha_1)^{-1} & & 0 \\ & \ddots & \\ 0 & & G(\alpha_n)^{-1} \end{bmatrix}.$$

And we have that $c \in \Gamma(L, G)$ if, and only if $Hc^T = \mathbf{0}$ if, and only if $XYc^T = \mathbf{0}$. Since X is a Vandermonde matrix, we have that the rank of XY is r, hence $d(\Gamma(L, G)) \geq r + 1$. Observing that H is a $r \times n$ matrix over $\mathbb{F}_{q^m}$, we can write H as a $rm \times n$ matrix over $\mathbb{F}_q$, and we have that the dimension of $\Gamma(L, G)$ is greater than or equal to $n - rm$. □

The next definitions are needed to understand the distinguisher attack.

Definition 10 (Schur Product)**.** [e.g. [18], Definition 2] Let $x, y \in \mathbb{F}_q^n$. We denote by the Schur product of x and y their componentwise product

$$x \star y = (x_1 y_1, \ldots, x_n y_n).$$

Remark 1. The Schur product is symmetric and bilinear.

Definition 11 (Schur Product of Codes and Square Code)**.** [e.g. [18], Definition 3] Let A, B be two codes of length n. Then we can define their Schur product to be the vector space spanned by all $a \star b$ with $a \in A$ and $b \in B$:

$$\langle A \star B \rangle = \langle \{ a \star b \mid a \in A, b \in B \} \rangle.$$

If $A = B$, then we call $\langle A \star A \rangle$ the square code of A and denote it by $\langle A^2 \rangle$.

Definition 12 (Schur Matrix)**.** [e.g. [39], Chapter 6, Definition 6.6.7] Let G be a $k \times n$ matrix, with rows g_i for $1 \leq i \leq k$. We denote by $S(G)$ the Schur matrix of G, which consists of the rows $g_i \star g_j$ for $1 \leq i \leq j \leq k$. Thus $S(G)$ is of the size $\frac{1}{2}(k^2 + k) \times n$.

We observe that by the remark above the Schur matrix of the generator matrix of a code is the generator matrix of the square code.

We want to take look at the dimension of the Schur product of codes, in particular at the dimension of square codes.

Proposition 5. [e.g. [18], Proposition 4] Let A, B denote two codes of length n. Then

$$\dim(\langle A \star B \rangle) \leq \dim(A)\dim(B).$$

Proof. Let $k = \dim(A)$ and $k' = \dim(B)$. Let the generator matrix of A have rows a_i for $1 \leq i \leq k$ and let the generator matrix of B have rows b_j for $1 \leq j \leq k'$. The generator matrix of $\langle A \star B \rangle$ has rows $a_i \star b_j$ for $1 \leq i \leq k$ and $1 \leq j \leq k'$. Thus the generator matrix of $\langle A \star B \rangle$ is of the size $kk' \times n$ and hence has at most rank

$$\min\{n, kk'\} \leq kk' = \dim(A)\dim(B).$$

□

Proposition 6. [e.g. [18], Proposition 4] Let A be a code of length n and dimension k, then

$$\dim(\langle A^2 \rangle) \leq \min\left\{ n, \binom{k+1}{2} \right\} \tag{1}$$

Proof. The generator matrix of the square code is given by the Schur matrix of the generator matrix of A. Since the Schur matrix is of size $\frac{1}{2}(k^2+k) \times n$ it has at most rank $\min\left\{n, \frac{1}{2}(k^2+k)\right\}$. □

The right hand side of (1) is considered as the maximal square code dimension. If A is a random code, it was shown in [17, 25, 10] that with high probability the square code of A will have maximal dimension. One can observe that for GRS codes this dimension is much smaller.

Proposition 7. [[18], Proposition 6] If $2k-1 < n$, then

$$\langle \mathrm{GRS}_{n,k}(\alpha,\beta)^2 \rangle = \mathrm{GRS}_{n,2k-1}(\alpha, \beta \star \beta).$$

This property can be adapted in the case $2k-1 \geq n$, by considering the dual of the GRS code, which is itself a GRS code. If $2k-1 \geq n$, then

$$\langle (\mathrm{GRS}_{n,k}(\alpha,\beta)^{\perp})^2 \rangle = \langle \mathrm{GRS}_{n,n-k}(\alpha,\gamma)^2 \rangle = \mathrm{GRS}_{n,2n-2k-1}(\alpha, \gamma \star \gamma).$$

For the proof of this proposition we will follow the idea of [18], Proposition 6.

Proof. Let c and c' be two codewords of the $\mathrm{GRS}_{n,k}(\alpha,\beta)$ code. Thus we can write

$$\begin{aligned} c &= (\beta_1 p(\alpha_1), \ldots, \beta_n p(\alpha_n)), \\ c' &= (\beta_1 q(\alpha_1), \ldots, \beta_n q(\alpha_n)). \end{aligned}$$

Where p and q are in $\mathbb{F}_q[x]$ and have degree strictly less than k. Then their Schur product has the following form.

$$\begin{aligned} c \star c' &= (\beta_1^2 p(\alpha_1)q(\alpha_1), \ldots, \beta_n^2 p(\alpha_n)q(\alpha_n)) \\ &= (\beta_1^2 r(\alpha_1), \ldots, \beta_n^2 r(\alpha_n)), \end{aligned}$$

where $\deg(r) \leq 2k-2$.

Hence if $2k-1 < n$, then the square code of a $\mathrm{GRS}_{n,k}(\alpha,\beta)$ code is a $\mathrm{GRS}_{n,2k-1}(\alpha,\beta)$ code, which has dimension $2k-1$.

□

3 Variants and Attacks

In the 70's McEliece was the first one to propose in [26] a cryptosystem using coding theory. The original system uses Goppa codes and is still unbroken. In the 80's Niederreiter proposed in [28] a dual cryptosystem to the McEliece system and was the first one to propose the use of GRS codes, which build an interesting coding family due to nice properties, such as MDS and fast decoding.

But in 1992 Sidelnikov and Shestakov showed in [37] that the Niederreiter system (and therefore also the McEliece system) is broken if GRS codes are used.

Bernstein, Lange and Peters [8] using Stern's algorithm [38] of decoding an arbitrary binary linear code by essentially brute force did demonstrate that the originally proposed McEliece system based on a [1024,512] Goppa code can be broken in reasonable time by modern computers. The attack [8] becomes however unfeasible for slightly larger code length.

Baldi, Bianchi, Chiaraluce, Rosenthal and Schipani proposed in [4] a variant of the McEliece system to reconsider GRS codes by changing the scrambling matrices. This thwarts the attack of Sidelnikov and Shestakov. Nevertheless Couvreur, Gaborit, Gauthier-Umaña, Otmani and Tillich attacked in [18, 13, 14] this proposal for some parameters. All these systems and attacks will now be stated.

3.1 McEliece System

McEliece proposed in [26] a cryptosystem based on the error correction capability of codes. The original proposal works as follows.

Choose $m \in \mathbb{N}$, $n = 2^m$, $t < \frac{n}{m}$ and a binary irreducible Goppa code Γ of length n, dimension $k \geq n - mt$, which can correct up to t errors. Γ has a generator matrix G of size $k \times n$. Choose a $k \times k$ matrix S with $\det(S) \neq 0$ and a $n \times n$ random permutation matrix P and compute $G' = SGP$.

$$\begin{aligned} \text{Public Key} &= (G', t), \\ \text{Private Key} &= (S, G, P). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_2^k$, and a random error vector $e \in \mathbb{F}_2^n$ with weight less than or equal to t, i.e. $\mathrm{wt}(e) \leq t$, then the cipher is computed as

$$y = xG' + e.$$

Decryption: Compute

$$yP^{-1} = xSG + eP^{-1},$$

then xSG is a codeword of Γ and since $\mathrm{wt}(eP^{-1}) \leq t$, we can apply the decoding algorithm for Goppa codes to get xS and by multiplying with the inverse of S we get the message x.

The McEliece code-based cryptosystem is still unbroken, but due to large key sizes not used in practice.

3.2 Niederreiter System

Niederreiter proposed in [28] a cryptosystem, which is similar to the McEliece system using the parity check matrix instead of the generator matrix and syndrome decoding for decryption. The Niederreiter cryptosystem works as follows.

Let $\mathbb{F}_q$ be a finite field. Let $1 \leq k < n \leq q$ be integers. Construct a $[n, k]$-linear code C, that can correct up to t errors and has an efficient decoding algorithm. C has a parity check matrix H of size $r \times n$, where $r = n - k$. Choose a $r \times r$ matrix S with $\det(S) \neq 0$ and a $n \times n$ random permutation matrix P and compute $H' = SHP$.

$$\begin{aligned} \text{Public Key} &= (H', t), \\ \text{Private Key} &= (S, H, P). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_q^n$ of weight less than or equal to t, i.e. $\mathrm{wt}(x) \leq t$, then the cipher is computed as

$$y^T = H'x^T.$$

Decryption: Compute

$$S^{-1}y^T = HPx^T = H(xP^T)^T.$$

Since $\mathrm{wt}(xP^T) \leq t$, we can apply syndrome decoding to get xP^T and by multiplying with the inverse of P^T we get the message x.

Observe that the Niederreiter system is dual to the McEliece system and they have equivalent security.

Proposition 8. [[21], page 272] The Niederreiter system and the McEliece system have equivalent security.

For the proof of this proposition we follow the idea of [21].

Proof. Let G be the $k \times n$ generator matrix of the linear code C and let H be the $r \times n$ parity check matrix. In the McEliece system the cipher is given by $c = mG' + e$. If we multiply on both sides with H'^T, then this becomes

$$z = cH'^T = mG'H'^T + eH'^T = eH'^T,$$

since $G'H'^T = 0$. This is the transposed of the Niederreiters encryption equation, thus if the Niederreiter system is broken, so is the McEliece system.

On the other hand, in the Niederreiter system the cipher is given by $c = H'x^T$, thus $c^T = xH'^T$. We can find a z of weight greater than or equal to t s.t. $c^T = zH'^T$. This z can be expressed as $z = mG' + x$ for some $m \in \mathbb{F}_q^k$. Which we observe to be the McEliece encryption equation. Therefore if the McEliece system is broken, so is the Niederreiter system. □

3.3 Attack of Sidelnikov and Shestakov

Niederreiter suggested to use the Niederreiter cryptosystem for C a GRS code, but in [37] Sidelnikov and Shestakov came up with an attack, which reveals the parity check matrix $\widetilde{H} = HP$, of a permutation equivalent code to the secret code, by just knowing the scrambled matrix SHP.

We will provide a short description of this attack, which can be found in [40].

Let $\widetilde{H} = HP$ be a $r \times n$ generator matrix of a $\mathrm{GRS}_{n,r}(\alpha, \beta)$ code and S a $n \times n$ invertible matrix. Given $M = S\widetilde{H}$ we want to recover β and α which determine the code completely. As first step we reconstruct α. Compute the echelon form of M, i.e.

$$\begin{pmatrix} 1 & & 0 & b_{1,r+1} & \cdots & b_{1,n} \\ & \ddots & & \vdots & & \vdots \\ 0 & & 1 & b_{r,r+1} & \cdots & b_{r,n} \end{pmatrix}.$$

we call the ith row of this matrix b_i and compute its associated polynomial f_{b_i}. Which is defined as follows. Let c be a codeword of a $\mathrm{GRS}_{n,r}(\alpha, \beta)$ code, then there exists a polynomial f_c associated to c, s.t.

$$c = (\beta_1 f_c(\alpha_1), \ldots, \beta_n f_c(\alpha_n)).$$

We observe that f_{b_i} has degree at most $r-1$, so it must be of the form

$$f_{b_i}(x) = c_{b_i} \prod_{\substack{j=1 \\ j \neq i}}^{r} (x - \alpha_j),$$

with $c_{b_i} \neq 0$. If we pick the rows, for example b_1 and b_2 and divide the entries of the first row by the entries of the second row, if they are nonzero, we get for $r+1 \leq j \leq n$

$$\frac{b_{1,j}}{b_{2,j}} = \frac{\beta_j f_{b_1}(\alpha_j)}{\beta_j f_{b_2}(\alpha_j)} = \frac{c_{b_1}(\alpha_j - \alpha_2)}{c_{b_2}(\alpha_j - \alpha_1)}.$$

By the Proposition 3 we can assume that $\alpha_1 = 0$ and $\alpha_2 = 1$. By this we know the $\frac{b_{1,j}}{b_{2,j}}$, hence the α_j can be reconstructed, if $\frac{c_{b_1}}{c_{b_2}}$ is guessed correctly. For the remaining $\alpha_3, \ldots, \alpha_n$ we take b_i instead of b_2 and we get for $3 \leq i \leq r$

$$\frac{b_{1,j}}{b_{i,j}}(\alpha_j - \alpha_1) = \frac{c_{b_1}}{c_{b_i}}(\alpha_j - \alpha_i).$$

By solving the system of linear equations one gets α.

The second step is to recover β.

Let $c = (c_1, \ldots, c_{r+1})$ be a nontrivial solution to $M'c^T = \mathbf{0}$, where M' denotes the $r \times (r+1)$ matrix consisting of the $r+1$ leftmost columns of M. Let H' be the $r \times (r+1)$ matrix consisting of the $r+1$ leftmost columns of the unknown $\widetilde{H}$, then $M' = SH'$, we know that $H'c^T = \mathbf{0}$. Hence

$$\begin{pmatrix} c_1 & \ldots & c_{r+1} \\ c_1\alpha_1 & \ldots & c_{r+1}\alpha_{r+1} \\ \vdots & & \vdots \\ c_1\alpha_1^{r-1} & \ldots & c_{r+1}\alpha_{r+1}^{r-1} \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_{r+1} \end{pmatrix} = 0.$$

By the Proposition 3 we can assume that $\beta_1 = 1$, then the solution is uniquely determined, and H' is completely known.
To compute the remaining β_i's we consider G'' the matrix consisting of the first r columns of H' and M'' the matrix consisting of the first r columns of M. Then $S = M''(H'')^{-1}$ and $\widetilde{H} = S^{-1}M$ and we can compute the remaining β_i's.

The attack of Sidelnikov and Shestakov uses the vulnerability that the public code is permutation equivalent to a GRS code.

This allows us to take away the influence of the scrambling matrix S. Then knowing the permutation equivalent code to the secret code is enough to recover the message. To avoid this attack, the public matrix should not be permutation equivalent to the secret code.

3.4 BBCRS Scheme

Baldi, Bianchi, Chiaraluce, Rosenthal and Schipani presented in [4] a variant of the McEliece cryptosystem, proposing to use GRS codes as secret code. To hide the algebraic structure of the secret code, they use instead of the permutation matrix the sum $T + R$, where T is a sparse matrix of row weight m and R is a matrix of rank z. This thwarts the attack of Sidelnikov and Shestakov [37].

We will denote this variant throughout this thesis by the BBCRS scheme. We will present the BBCRS scheme, following [4] in both the McEliece and the Niederreiter version.

The BBCRS scheme in the McEliece version works as follows.

Let $\mathbb{F}_q$ be a finite field. Let $1 \leq k < n \leq q$ be integers. Choose a systematic $k \times n$ generator matrix G of a linear block code C over $\mathbb{F}_q$, with an efficient decoding algorithm and the ability to correct up to t errors.

Choose a $k \times k$ invertible matrix S. Let R be a $n \times n$ matrix of rank z, obtained as follows. Let $\omega \in \mathbb{N}$ and choose $a_1, \ldots, a_\omega$ and $b_1, \ldots, b_\omega$ to be $z \times n$ matrices, where $z \leq n$. Define $a = \sum_{i=1}^{\omega} a_i$ and

$$R = \begin{pmatrix} a_1 \\ \vdots \\ a_\omega \end{pmatrix}^T \cdot \begin{pmatrix} b_1 \\ \vdots \\ b_\omega \end{pmatrix}.$$

In this proposal they focus on two cases, both with $\omega = 2$:

1. $a_1 = a, a_2 = \mathbf{0}$, where $\mathbf{0}$ stands for the all zero matrix, which means

$$R = \begin{pmatrix} a \\ \mathbf{0} \end{pmatrix}^T \cdot \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

2. $b_1 = b, b_2 = \mathbf{1} + b$, where $\mathbf{1}$ stands for the all one matrix, hence

$$R = \begin{pmatrix} a_1 \\ a_2 \end{pmatrix}^T \cdot \begin{pmatrix} b \\ \mathbf{1} + b \end{pmatrix}.$$

Choose T an $n \times n$ invertible sparse matrix over $\mathbb{F}_q$ with average row and column weight m. So, if $m \in \mathbb{Z}$, then T is the sum of m generalized permutation

matrices, where the nonzero entries do not overlap. If $m \in \mathbb{Q}$, then T is almost a regular matrix with weight $\lceil m \rceil$ or $\lfloor m \rfloor$. R and T are chosen s.t. $Q = R + T$ is an $n \times n$ invertible matrix. Define $t_{\text{pub}} = \lfloor \frac{t}{m} \rfloor$. The error vector e can have further constraints, i.e.

$$ae^T = 0. \tag{2}$$

This would require a to be public. In [4] it is shown how to avoid this possible weakness.

Compute $G' = S^{-1}GQ^{-1}$.

$$\begin{aligned} \text{Public Key} &= (G', t_{\text{pub}}), \\ \text{Private Key} &= (S, G, Q). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_q^k$, and a random error vector $e \in \mathbb{F}_q^n$ with weight less than or equal to t_{pub}, i.e. $\text{wt}(e) \leq t_{\text{pub}}$. Then the cipher is computed as

$$y = xG' + e.$$

Decryption: Compute

$$yQ = xS^{-1}G + eQ,$$

where $eQ = e(R + T) = eR + eT$. And by (2) we have

$$eR = \begin{cases} \mathbf{0} & \text{if } a_2 = \mathbf{0}, a = a_1, \\ ea_2^T\mathbf{1} & \text{if } b_1 = b, b_2 = \mathbf{1} + b. \end{cases}$$

One can show that eQ can be reduced to eT, which is clear in the first case. To achieve this in the second case the receiver should know the value $ea_2^T = \gamma \in \mathbb{F}_q$, so one needs to test at most q choices. Observe that $xS^{-1}G$ is a codeword of C and since $\text{wt}(eT) \leq t$, we can apply the decoding algorithm of C to get xS^{-1} and by multiplying with S we get the message x.

It is reasonable to look at the parameters $z = 1, m = 1$, for key size and complexity reason: If m increases, also the key size increases, if z increases, the complexity of the decoding algorithm increases.

We give a short overview of the BBCRS scheme for the choice $z = 1, m = 1$.

$$\begin{aligned} G &= k \times n \text{ generator matrix of GRS code}, \\ T &= n \times n \text{ permutation matrix}, \\ R &= n \times n \text{ rank 1 matrix}, \ R = \alpha^T\beta, \\ Q &= n \times n \text{ invertible matrix}, \ Q = R + T, \\ S &= k \times k \text{ invertible matrix}. \end{aligned}$$

Compute: $G' = S^{-1}GQ^{-1}$ and $t_{\text{pub}} = t = \lfloor \frac{n-k}{2} \rfloor$.

$$\begin{aligned} \text{Public Key} &= (G', t), \\ \text{Private Key} &= (G, T, R, Q, S). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_q^k$ and a random error vector $e \in \mathbb{F}_q^n$, s.t. $\text{wt}(e) \leq t$. Compute the cipher as $y = xG' + e$.

Decryption: Guess the value of eR. Since $eR = e\alpha^T\beta = \gamma\beta$ for some $\gamma \in \mathbb{F}_q$, it is enough to find γ and thus we test at most q values. Then compute $y' = yQ - eR = xS^{-1}G + eT$. Since $\mathrm{wt}(eT) \leq t$ by the decoding algorithm of the GRS code we get xS^{-1} and hence x.

Now we will state the BBCRS scheme in the Niederreiter version.

Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ integers. Let H be a $r \times n$ parity check matrix of a linear block code over $\mathbb{F}_q$, where $r = n - k$. Choose a $r \times r$ invertible matrix S and R, T, Q as in the McEliece version.

Compute $H' = S^{-1}HQ^T$.

$$\begin{aligned} \text{Public Key} &= (H', t_{\text{pub}}), \\ \text{Private Key} &= (S, H, Q). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_q^n$ of weight less than or equal to t_{pub}, i.e. $\mathrm{wt}(x) \leq t_{\text{pub}}$. Then the cipher is computed as

$$y = H'x^T.$$

Decryption: Compute

$$y' = Sy = HQ^Tx^T = H(xQ)^T,$$

where xQ can again be reduced to xT, hence $y' = HT^Tx^T$. Observe that HT^Tx^T is a codeword and since $\mathrm{wt}(T^Tx^T) \leq t$, by syndrome decoding we get T^Tx^T. Thus by multiplying with the inverse of (T^T) we get the message x.

Please observe that the new proposal, presented in [9] is a special case of the BBCRS scheme, since it takes the parameters $z = 0$ and $m = 2$.

3.5 Distinguisher Attack

Couvreur, Gaborit, Gauthier-Umaña, Otmani and Tillich presented in [18, 13] for some parameters a distinguisher attack on the BBCRS scheme [4]. They observed that the public matrix of the BBCRS scheme has unusual small square code dimension.

The idea of the attack relies on finding a subcode of the public code of codimension 1. This is also a subcode of a GRS code, which is permutation equivalent to the secret code. One finds this subcode by using the small square code dimension. The square code of the subcode is with high probability the square code of the GRS code. The attack recovers this GRS code, which is permutation equivalent to the secret code and with this one can recover the message.

Since the attack in [18, 13] uses the square code dimension of the public code, to distinguish if the secret code was a GRS code and not chosen randomly, we will call this attack the distinguisher attack throughout this thesis.

The attack assumes that $2k + 2 < n$. If this is not the case, the same attack can be applied to the dual of the public code, under the assumption $2k > n + 2$. Hence the attack has a gap for $\frac{n-2}{2} \leq k \leq \frac{n+2}{2}$.

We will now provide a description of the distinguisher attack, following [18, 13], under the assumption that $2k + 2 < n$.
We denote by $\mathcal{C}_{\text{pub}}$ the public code of the BBCRS scheme and by $\mathcal{C}_{\text{sec}}$ the secret code, which is a GRS code. Recall that in the BBCRS scheme the scrambling matrix Q is given by $Q = R + \Pi$, where R is a matrix of rank 1 and Π is a $n \times n$ permutation matrix.

Define

$$\mathcal{C} = \mathcal{C}_{\text{sec}}\Pi^{-1}.$$

Since $\mathcal{C}$ is permutation equivalent to a GRS code, it is again a GRS code, i.e.

$$\mathcal{C} = \text{GRS}_k(x, y).$$

Let $a, b \in \mathbb{F}_q^n$, s.t.

$$R\Pi^{-1} = b^T a.$$

Define

$$\lambda = -\frac{1}{1 + a \cdot b} b.$$

Then the public code and $\mathcal{C}$ are connected by the following property.

Lemma 3. [[18], Lemma 1] For any c in $\mathcal{C}_{\text{pub}}$, there exists p in $\mathcal{C}$, such that

$$c = p + (p \cdot \lambda)a. \tag{3}$$

For the proof of this lemma we follow the idea of [18], Lemma 1.

Proof. We first want to write $\mathcal{C}_{\text{pub}}$ in terms of $\mathcal{C}$.

$$\mathcal{C} = \mathcal{C}_{\text{sec}}\Pi^{-1} = \mathcal{C}_{\text{pub}}Q\Pi^{-1} = \mathcal{C}_{\text{pub}}\left(\Pi + R\right)\Pi^{-1} = \mathcal{C}_{\text{pub}}\left(I + R\Pi^{-1}\right) = \mathcal{C}_{\text{pub}}P,$$

for

$$P = I + R\Pi^{-1} = I + b^T a.$$

Hence

$$\mathcal{C}_{\text{pub}} = \mathcal{C}P^{-1}.$$

Now we want to compute P^{-1}. We use the following lemma of [27].

Lemma 4. [[27], page 68] Let A and $A + B$ be invertible matrices and B has rank 1. Then

$$(A + B)^{-1} = A^{-1} - \frac{1}{1 + \text{tr}\left(BA^{-1}\right)} A^{-1}BA^{-1},$$

where $\text{tr}(\cdot)$ denotes the trace and $\text{tr}(BA^{-1}) \neq -1$.

We apply this on $A = I$, $B = b^T a$ and since $\text{tr}(b^T a) = a \cdot b$ we get

$$P^{-1} = I - \frac{1}{1 + a \cdot b} b^T a.$$

Since $\mathcal{C}_{\text{pub}} = \mathcal{C}P^{-1}$, for each $c \in \mathcal{C}_{\text{pub}}$, there exists a $p \in \mathcal{C}$, s.t.

$$\begin{aligned} c &= pP^{-1} = p\left(I - \frac{1}{1+a\cdot b} b^T a\right) \\ &= p - \frac{1}{1+a\cdot b} pb^T a \\ &= p + \left(p\cdot b\left(-\frac{1}{1+a\cdot b}\right)\right) a \\ &= p + (p\cdot\lambda)a. \end{aligned}$$

□

If $\lambda \in \mathcal{C}^{\perp}$, then $p \cdot \lambda = 0$ and the above lemma yields that $\mathcal{C}_{\text{pub}} = \mathcal{C} = \text{GRS}_k(x, y)$.

Hence we can apply the attack of Sidelnikov and Shestakov [37] to the public code to reveal x and y. This completely defines the GRS code and we can decode any cipher by the decoding algorithm of $\text{GRS}_k(x, y)$. Hence we can assume that $\lambda \notin \mathcal{C}^{\perp}$. Define the following code

$$\mathcal{C}_{\lambda^{\perp}} = \mathcal{C} \cap \langle\lambda\rangle^{\perp},$$

where $\langle\lambda\rangle$ denotes the vector space spanned by λ. This code consists of the codewords $c = p + (p \cdot \lambda)a$ with $p \in \mathcal{C}$, s.t. $p \cdot \lambda = 0$. Therefore this code is a subcode of $\mathcal{C}_{\text{pub}}$ and also of $\mathcal{C}$. Observe that $\mathcal{C}_{\text{pub}}$ and $\mathcal{C}$ are both of dimension k and the subcode $\mathcal{C}_{\lambda^{\perp}}$ has dimension $k - 1$.

The public code of the BBCRS scheme has an unusual small square code dimension, since

Proposition 9. [[18], Proposition 4] $\dim(\langle\mathcal{C}^2_{\text{pub}}\rangle) \leq 3k - 1$.

For the proof of this proposition we follow the idea of [13], Proposition 14.

Proof. Let b_i for $1 \leq i \leq k$ be a basis of $\mathcal{C}_{\text{pub}}$, s.t. b_i for $1 \leq i \leq k-1$ is a basis for the subcode $\mathcal{C}_{\lambda^{\perp}}$. Since $\mathcal{C}_{\lambda^{\perp}}$ is also a subcode of $\mathcal{C}$, which we recall is a GRS code of dimension k, hence by Proposition 7, we have that

$$\dim(\langle\mathcal{C}^2\rangle) = 2k - 1,$$

therefore also

$$\dim(\langle\mathcal{C}^2_{\lambda^{\perp}}\rangle) \leq 2k - 1.$$

$\langle\mathcal{C}^2_{\text{pub}}\rangle$ is generated by $b_i \star b_j$ for $1 \leq i, j \leq k$ and $\langle\mathcal{C}^2_{\lambda^{\perp}}\rangle$ is generated by $b_i \star b_j$ for $1 \leq i, j \leq k-1$. So comparing $\langle\mathcal{C}^2_{\text{pub}}\rangle$ to $\langle\mathcal{C}^2_{\lambda^{\perp}}\rangle$, only $b_i \star b_k$ for $1 \leq i \leq k$ are possibly not in $\langle\mathcal{C}^2_{\lambda^{\perp}}\rangle$, which span a space of dimension k. Therefore

$$\dim(\langle\mathcal{C}^2_{\text{pub}}\rangle) \leq 2k - 1 + k = 3k - 1.$$

□

We want to find the large subcode $\mathcal{C}_{\lambda^{\perp}}$, for this we use the following observation.

Let $g_1, \ldots, g_k$ be basis of $\mathcal{C}_{\text{pub}}$ and take random other elements z_1, z_2, z_3 from $\mathcal{C}_{\text{pub}}$. Then define

$$\mathcal{B} = \{z_i \star g_j \mid 1 \leq i \leq 3,\ 1 \leq j \leq k\}. \tag{4}$$

We have the following proposition about the dimension of $\mathcal{B}$.

Proposition 10. [[18], Proposition 5] Let $\mathcal{B}$ be defined as in (4). Then $\dim(\mathcal{B}) \leq 3k - 3$.

For the proof of this proposition we follow the idea of [13], Proposition 15.

Proof. Since the g_i for $1 \leq i \leq k$ are a basis of $\mathcal{C}_{\text{pub}}$, we can write the z_i in terms of g_i, i.e.

$$z_i = \sum_{1 \leq j \leq k} a_{ij} g_j.$$

There exist the following three independent relations between the $z_i \star g_j$:

$$\begin{aligned}
\sum_{1 \leq j \leq k} a_{2j} z_1 \star g_j - \sum_{1 \leq j \leq k} a_{1j} z_2 \star g_j &= z_1 \star z_2 - z_2 \star z_1 = 0, \\
\sum_{1 \leq j \leq k} a_{3j} z_1 \star g_j - \sum_{1 \leq j \leq k} a_{1j} z_3 \star g_j &= z_1 \star z_3 - z_3 \star z_1 = 0, \\
\sum_{1 \leq j \leq k} a_{3j} z_2 \star g_j - \sum_{1 \leq j \leq k} a_{2j} z_3 \star g_j &= z_2 \star z_3 - z_3 \star z_2 = 0.
\end{aligned}$$

For these relations to be independent, we refer to [13], Proposition 15. □

If we have picked the random elements z_i in $\mathcal{C}_{\lambda^\perp}$, then we have a smaller square code dimension.

Proposition 11. [[18], Proposition 6] Let $\mathcal{B}$ be defined as in (4). If $z_i \in \mathcal{C}_{\lambda^\perp}$ for $1 \leq i \leq 3$, then

$$\dim(\mathcal{B}) \leq 2k + 2.$$

For the proof of this proposition we follow the idea of [18], Proposition 6.

Proof. Assume $z_i \in \mathcal{C}_{\lambda^\perp}$. Let g_j for $1 \leq j \leq k$ build a basis of $\mathcal{C}_{\text{pub}}$. By Lemma 3 there exists a $p_j \in \mathcal{C}$ for every g_j, s.t.

$$g_j = p_j + (\lambda \cdot p_j)a.$$

If we compute the Schur product of z_i and g_j we therefore get

$$\begin{aligned}
z_i \star g_j &= z_i \star (p_j + (\lambda \cdot p_j)a) \\
&= z_i \star p_j + (\lambda \cdot p_j) z_i \star a.
\end{aligned}$$

By this we have that

$$\langle z_i \star g_j \rangle \subset \langle \mathcal{C}^2 \rangle + \langle z_1 \star a \rangle + \langle z_2 \star a \rangle + \langle z_3 \star a \rangle.$$

Since $\mathcal{C}$ is a GRS code of dimension k, we have by Proposition 7 that $\langle \mathcal{C}^2 \rangle$ has dimension $2k - 1$. The vector space generated by $z_i \star a$ has dimension at most 3. Therefore we get

$$\dim(\mathcal{B}) \leq 2k - 1 + 3 = 2k + 2.$$

□

Thus by taking the basis $g_1, \ldots, g_k$ of $\mathcal{C}_{\text{pub}}$ and taking random other elements z_1, z_2, z_3 from $\mathcal{C}_{\text{pub}}$, and testing if the set $\mathcal{B}$, generated by these elements as in (4) has dimension less than or equal to $2k+2$, we can add the elements z_1, z_2, z_3 to the basis of $\mathcal{C}_{\lambda^\perp}$ and repeat this step to find a basis $z_1, \ldots, z_{k-1}$ of $\mathcal{C}_{\lambda^\perp}$. A complete description of this algorithm can be found in [18], Algorithm 1.

Pellikaan *et al.* showed in [23] that a large subcode of a GRS code has with high probability as square code the square code of the GRS code.

We can apply this on the code $\mathcal{C}_{\lambda^\perp}$, which is a large subcode of $\text{GRS}_{n,k}(x,y)$. By computing its square code we get with high probability $\text{GRS}_{n,2k-1}(x, y \star y)$. We can apply the attack of Sidelnikov and Shestakov to recover x and $y \star y$ and therefore also x and y, which completely determine $\mathcal{C}$, which we recall is permutation equivalent to the secret GRS code.

As last step we want to find a, λ, which satisfy (3) and $a \cdot \lambda \neq -1$. We take

$$\begin{aligned} a &= a_0 \in (\mathcal{C}_{\text{pub}}^\perp \cap \mathcal{C}^\perp)^\perp \backslash \mathcal{C}, \\ b_0 &\in (\mathcal{C}_{\text{pub}} \cap \mathcal{C})^\perp \backslash \mathcal{C}^\perp \text{ s.t. } a_0 \cdot b_0 = 0, \\ r_1 &\in \mathcal{C}_{\text{pub}}^\perp \backslash \mathcal{C}^\perp, \text{ s.t. } a_0 \cdot r_1 \neq 0, \\ p_1 &\in \mathcal{C} \backslash \mathcal{C}_{\text{pub}}, \\ \gamma &= \frac{-(p_1 \cdot r_1)}{(b_0 \cdot p_1)(a_0 \cdot r_1)}, \\ \lambda &= \gamma b_0. \end{aligned}$$

Observe that $b_0 \cdot p_1 \neq 0$ since $p_1 \in \mathcal{C} \backslash \mathcal{C}_{\text{pub}}$ and since

$$\mathcal{C}_{\text{pub}} \cap \mathcal{C} = \left\{ p \in \mathcal{C} \;\middle|\; p \cdot \lambda = 0 \right\}.$$

With this pair one can recover the message. Assume that we received $z = c + e$, where $c \in \mathcal{C}_{\text{pub}}$. We know that there exists a $p \in \mathcal{C}$, s.t. $c = p + (\lambda \cdot p)a$. We compute for all $\alpha \in \mathbb{F}_q$ the value $z + \alpha a$. If we have chosen the correct $\alpha = -\lambda \cdot p$, then $z + \alpha a = p + e$ and by the decoding algorithm of $\mathcal{C}$ we get the message.

Let us recap the distinguisher attack.

1. Take a basis g_j of $\mathcal{C}_{\text{pub}}$ and compute $\{z_i \star g_j\}$ for $z_1, z_2, z_3 \in \mathcal{C}_{\text{pub}}$. If the dimension of this is smaller than $2k+2$, then $z_i \in \mathcal{C}_{\lambda^\perp}$. This way we build a basis of $\mathcal{C}_{\lambda^\perp}$.

2. Compute the square code of $\mathcal{C}_{\lambda^\perp}$, which will be with high probability $\text{GRS}_{n,2k-1}(x, y \star y)$. We apply the attack of Sidelnikov and Shestakov to get $\mathcal{C} = \text{GRS}_{n,k}(x,y)$.

3. We want to find a, λ, s.t. for all $c \in \mathcal{C}_{\text{pub}}$ there exists a $p \in \mathcal{C}$, s.t. $c = p + (p \cdot \lambda)a$.

4. Knowing a, λ and the GRS code $\mathcal{C}$ we can decode any cipher.

3.6 Extended Distinguisher Attack

Couvreur, Gaborit, Gauthier-Umaña, Otmani and Tillich presented in [14] a second attack on the BBCRS scheme, which extends the choice of the the parameters to $z = 1,\ m \leq 1 + \frac{k}{n} + \mathcal{O}(\frac{1}{\sqrt{n}})$. Where we recall that the scrambling matrix Q is given by $Q = R + T$, where R is a matrix of rank 1 and T is a $n \times n$ matrix of weight m.

In the extended distinguisher attack it is observed that in the Niederreiter version of the BBCRS scheme puncturing the public code gives a small square code dimension. This helps to detect the weights of the rows of T and to reduce the case $z = 1,\ m \leq 1 + \frac{k}{n} + \mathcal{O}(\frac{1}{\sqrt{n}})$ to the case $z = 1$ and $m = 1$, for which one can apply the distinguisher attack [18, 13]. We first need the following notations. Since $m \leq 1 + \frac{k}{n} < 2$, we can assume that T has only rows of weight 1 and rows of weight 2.

J_1 denotes the set of positions which correspond to rows of T of weight 1. For a row i in J_1, we denote by $j(i)$ the unique column of T, s.t. $T_{i,j(i)} \neq 0$.

J_2 denotes the set of positions which correspond to rows of T of weight 2. For a row i in J_2, we denote by $j(i) = \{j_1, j_2\}$ the two columns of T, s.t. $T_{i,j_1} \neq 0$ and $T_{i,j_2} \neq 0$.

We want to recall the definitions of shortening and puncturing.

Definition 13. [e.g. in [22], Chapter 1, page 28] Let C be a $[n, k]$ linear block code over $\mathbb{F}_q$. Let $I \subset \{1, \ldots, n\}$. Then we define the punctured code $P_I(C)$ and the shortened code $S_I(C)$ as

$$\begin{aligned} P_I(C) &= \left\{(c_i)_{i \notin I} \;\middle|\; c \in \mathcal{C}\right\}, \\ S_I(C) &= \left\{(c_i)_{i \notin I} \;\middle|\; \exists c \in \mathcal{C},\ \text{s.t. } c_i = 0\ \forall i \in I\right\}. \end{aligned}$$

We have the following observations concerning the square code dimension of the shortened code.

Remark 2. [[14], page 9] Let C be a $[n, r]$ linear block code, and $I \subset \{1, \ldots, n\}$. Then

$$\dim\left(\langle S_I(C)^2\rangle\right) \leq \min\left\{n - |I|, \binom{r_I + 1}{2}\right\},$$

where $r_I = \dim(S_I(C))$, which is in general $r - |I|$.

This follows directly from Proposition 6 and observing that $S_I(C)$ is a $[n - |I|, r_I]$ code.

If C is a random $[n, r]$ code, we expect the square code dimension of the shortened code to reach this maximal dimension, as before we refer to [17, 10, 25]. Whereas for the public code of the BBCRS scheme in the Niederreiter version, this dimension is much smaller, therefore with the next proposition one can distinguish if a GRS code or a random code was used as secret code.

Proposition 12. [[14], Proposition 4] Let $\mathcal{C}_{\text{pub}}$ be the $[n, r]$ public code of the BBCRS scheme and $I \subset J_1$. Then

$$\dim\left(\langle S_I(\mathcal{C}_{\text{pub}})^2\rangle\right) \leq 3r + |J_2| - 3|I| - 1. \tag{5}$$

For the proof of this proposition we first need some other lemmata.

Lemma 5. [[14], Lemma 5] Let $I_1 \subset J_1$, s.t. $\mid I_1 \mid = s \leq r$. Then there exist $a, u, v \in \mathbb{F}_q^{n-s-\left|J_2\right|}$, s.t.

$$P_{J_2}\left(S_{I_1}\left(\mathcal{C}_{\text{pub}}\right)\right) \subseteq \varepsilon + \langle a \rangle, \tag{6}$$

where ε is a subcode of a $\text{GRS}_{n,r-s}(u,v)$ code.

For the proof of this lemma we refer the interested reader to [14], Lemma 5.

Remark 3. Let A and B be $[n,k]$ linear codes over $\mathbb{F}_q$, then since

$$\langle (A+B)^2 \rangle = \langle A^2 \rangle + \langle B^2 \rangle + \langle A \star B \rangle,$$

we have that

$$\dim\left(\langle (A+B)^2 \rangle\right) \leq \dim\left(\langle A^2 \rangle\right) + \dim\left(\langle B^2 \rangle\right) + \dim\left(\langle A \star B \rangle\right). \tag{7}$$

Lemma 6. [[14], Lemma 6] Let C be an $[n,k]$ linear code over $\mathbb{F}_q$, let $I \subset \{1,\ldots,n\}$. Then

$$\dim\left(\langle C^2 \rangle\right) \leq \dim\left(\langle P_I(C)^2 \rangle\right) + \mid I \mid. \tag{8}$$

For the proof of this lemma we follow the idea of [14], Lemma 6.

Proof. Let $S(I)$ be the code which consists of all $x \in \mathbb{F}_q^n$, s.t. $x_i = 0 \ \forall i \notin I$. Therefore $S(I)$ is an $[n, \mid I \mid]$ code over $\mathbb{F}_q$. Let $E(P_I(C))$ be the code, which extends $P_I(C)$ by zeros to length n. Hence $C \subseteq E(P_I(C)) \oplus S(I)$. We apply (7), and get

$$\begin{aligned} \dim(\langle C^2 \rangle) &= \dim\left(\langle (E(P_I(C)) + S(I))^2 \rangle\right) \\ &\leq \dim\left(\langle E(P_I(C))^2 \rangle\right) + \dim\left(\langle S(I)^2 \rangle\right) + \dim\left(\langle E(P_I(C)) \star S(I) \rangle\right). \end{aligned}$$

Now observe that $E(P_I(C)) \star S(I) = \{0\}$, $\dim\left(\langle E(P_I(C))^2 \rangle\right) = \dim\left(\langle P_I(C)^2 \rangle\right)$ and $\dim\left(\langle S(I)^2 \rangle\right) = \mid I \mid$. Hence we get

$$\dim\left(\langle C^2 \rangle\right) \leq \dim\left(\langle P_I(C)^2 \rangle\right) + \mid I \mid.$$

□

For the proof of the Proposition 12 we follow the idea of [14], Proposition 4.

Proof. We apply Lemma 5 to $I_1 = I \subset J_1$, $s = \mid I \mid$ and by (6) we get

$$P_{J_2}\left(S_I\left(\mathcal{C}_{\text{pub}}\right)\right) \subseteq \varepsilon + \langle a \rangle,$$

where ε is a subcode of a $\text{GRS}_{n,r-\left|I\right|}(u,v)$ code. We apply the Remark 3 to $A = \langle a \rangle$ and $B = \varepsilon$ to get

$$\begin{aligned} \dim\left(\left\langle P_{J_2}\left(S_I\left(\mathcal{C}_{\text{pub}}\right)\right)^2 \right\rangle\right) &\leq \dim\left(\langle \varepsilon^2 \rangle\right) + \dim\left(\langle \langle a \rangle \star \varepsilon \rangle\right) + \dim\left(\langle \langle a \rangle^2 \rangle\right) \\ &\leq \dim\left(\langle \varepsilon^2 \rangle\right) + \dim\left(\varepsilon\right) + 1. \end{aligned}$$

Since ε is a subcode of a GRS code of dimension $r-\mid I \mid$, ε has dimension at most $r-\mid I \mid -1$. By Proposition 7 we know that

$$\dim\left(\langle\varepsilon^2\rangle\right) \leq \dim\left(\langle \mathrm{GRS}_{n,r-|I|}(u,v)^2\rangle\right) \leq 2r-2\mid I \mid -1.$$

Therefore we get

$$\begin{aligned}\dim\left(\langle P_{J_2}\left(S_I\left(\mathcal{C}_{\mathrm{pub}}\right)\right)^2\rangle\right) &\leq 2r-2\mid I \mid -1+r-\mid I \mid -1+1\\ &= 3r-3\mid I \mid -1.\end{aligned}$$

Now we apply Lemma 6 to $C = S_I\left(\mathcal{C}_{\mathrm{pub}}\right)$ and $I = J_2$. We get by (8) the claim, since

$$\begin{aligned}\dim\left(\langle S_I(\mathcal{C}_{\mathrm{pub}})^2\rangle\right) &\leq \dim\left(\langle P_{J_2}\left(S_I\left(\mathcal{C}_{\mathrm{pub}}\right)\right)^2\rangle\right)+\mid J_2 \mid\\ &\leq 3r-3\mid I \mid -1+\mid J_2 \mid.\end{aligned}$$

□

By using similar results, one can determine which positions come from rows of weight 1 or 2. We puncture $S_I(\mathcal{C}_{\mathrm{pub}})$ in a position $i \notin I$, if the dimension of the square code gets smaller, then we have that the position i is of degree 2. Since then

$$\dim\left(\langle S_I(\mathcal{C}_{\mathrm{pub}})^2\rangle\right) = \dim\left(\langle P_i\left(S_I(\mathcal{C}_{\mathrm{pub}})\right)^2\rangle\right)+1. \tag{9}$$

Whereas for a position of degree 1 one expects that the dimension stays the same.

Therefore to compute the set J_2 we choose random subsets $I_1,\ldots,I_s$ and test if (5) is satisfied. Then we define $J_2(i)$ to be the set of positions, for which (9) is satisfied for $1 \leq i \leq s$. Then we set J_2 to be $J_2(1)\cup\cdots\cup J_2(s)$.

Now we want to transform the degree 2 positions into positions of degree 1, by linear combinations of columns of weight 1. For a position $i_1 \in J_1$, we compute the set of i_2 which consists of positions in J_2, s.t. $j(i_1) \in j(i_2)$. We go through all $\alpha \in \mathbb{F}_q^\times$ and multiply the public code to a matrix, which has on the diagonal all ones and α in the entry (i_1,i_2). For the correct α this new code $\mathcal{C}$ is

$$\mathcal{C} = \mathcal{C}_{\mathrm{sec}}(T'+R')^T,$$

where the row i_2 of T' has now weight 1. We iterate this for $\mathcal{C}$ instead of $\mathcal{C}_{\mathrm{pub}}$ and $J_1\cup\{i_2\}$ instead of J_1. If we are successful, we end up with a code, which is of the form

$$\mathcal{C} = \mathcal{C}_{\mathrm{sec}}(\widetilde{T}+\widetilde{R})^T,$$

where $\widetilde{T}$ is a matrix of weight 1 and $\widetilde{R}$ a matrix of rank 1, so we can apply the distinguisher attack of the case $m=1, z=1$. For a more detailed procedure, we refer to [14].

The attack can be summarized as follows.

1. The public matrix H' is assumed to have columns of weight 1 and 2, which are detected, using the square code dimension when puncturing the shortened public code.
2. We transform the columns of weight 2 into columns of weight 1 by linear combinations of the columns of weight 1.
3. Now one can apply the distinguisher attack [13, 18].

3.7 ISD Attack

One possibility to attack a code-based cryptosystem is information set decoding (ISD), which means to decode a random code without exploiting any structural property of the code, hence it is non-polynomial in the dimension of the code. In [31] they provide a generalization of Stern's algorithm [38]. In the following we shortly explain the idea of the ISD attack.

Let C be an $[n, k]$ code over $\mathbb{F}_q$, and let G denote the generator matrix of C. For a nonempty subset $I \subset \{1, \ldots, n\}$, we denote by G_I the matrix consisting of the columns indexed by I. If we take I s.t. $\mid I \mid = k$ and G_I is invertible, then we call the I-indexed entries of a codeword the information symbols and I an information set. Let $c \in \mathbb{F}_q^n$ with distance t from the code. The ISD attack finds a vector e, which can be written as $e = c + mG$, for some $m \in \mathbb{F}_q^k$ and $\mathrm{wt}(e) = t$.

We provide the algorithm of [31]. Let C be an $[n, k]$ code over $\mathbb{F}_q$, and let G denote the generator matrix of C. Assume k is even. Let $0 \leq p \leq t$ and $0 \leq l \leq r = n - k$ be integers. We have as input the generator matrix G, and the integers $k < n \leq q$ and $c \in \mathbb{F}_q^n$ with distance t from the code.

1. Choose an information set I.
2. Replace c by $c - c_I G_I^{-1} G$.
3. Choose a uniform random subset $X \subset I$, with $\mid X \mid = \frac{k}{2}$ and set $Y = I \backslash X$.
4. Choose a uniform random subset $Z \subset \{1, \ldots, n\} \backslash I$, with $\mid Z \mid = l$.
5. For any $A = \{a_1, \ldots, a_p\} \subset X$ define

$$V_A = \left\{ c - \sum_{i=1}^{p} m_i g_{a_i} \;\middle|\; m = (m_1, \ldots, m_p) \in \left(\mathbb{F}_q^{\times}\right)^p \right\}.$$

   For each $v \in V_A$ compute $v(Z) \in \mathbb{F}_q^l$, which are the entries of v indexed by Z.

6. For any $B = \{b_1, \ldots, b_p\} \subset Y$ define

$$V_B = \left\{ c - \sum_{j=1}^{p} m'_j g_{b_j} \;\middle|\; m' = (m'_1, \ldots, m'_p) \in \left(\mathbb{F}_q^{\times}\right)^p \right\}.$$

For each $v' \in V_B$ compute $v'(Z) \in \mathbb{F}_q^l$, which are the entries of v' indexed by Z.

7. For each pair (A, B), s.t. there exists a pair of vectors $v = c - \sum_{i=1}^{p} m_i g_{a_i}$ and $v' = \sum_{j=1}^{p} m'_j g_{b_j}$ and $v(Z) = v'(Z)$, we compute $e = v - v'$. If $\mathrm{wt}(e) \leq t$ we are done, else we go back to 1.

These seven steps are one iteration of the algorithm. For the cost of one iteration of the algorithm, we need to introduce two new parameters, $1 < d \leq c < k$ which are introduced for speed up. If the choice of I in the first step does not lead to a vector e of weight less than or equal to t, then we reuse $k-c$ of the columns of G_I and choose c new linearly independent columns. The parameter d is for faster pivoting by precomputing the sum of the d first rows. By this we get that the cost of one iteration is, as computed in [31]:

$$\begin{aligned} w &= (n-1)\left((k-1)\left(1-\frac{1}{q^d}\right)+(q^d-d)\right)\frac{c}{d} \\ &+ \left(\left(\frac{k}{2}-p+1\right)+2\binom{k/2}{p}(q-1)^p\right)l \\ &+ \frac{q}{q-1}(t-2p+1)2p\left(1+\frac{q-2}{q-1}\right)\frac{\binom{k/2}{p}^2(q-1)^{2p}}{q^l}. \end{aligned}$$

Since we reuse $k-c$ columns of G_I the iterations of the algorithm are not independent. In [31] the number of needed iterations is computed as in [8], using a Markov chain with the $t+2$ states:

- 0: The chosen information set contains 0 errors.
- ...
- t: The chosen information set contains t errors.
- Done: The attack has succeeded.

For the new choice of the information set, we choose c positions of I and change them with c positions which are not in I. An iteration moves from state s to $s+a$ with the probability

$$\frac{\sum_i \binom{t-s}{i}\binom{n-k-t+s}{c-i}\binom{s}{a+i}\binom{k-s}{c-a-i}}{\binom{n-k}{c}\binom{k}{c}}.$$

In the state $2p$ we move to the state Done, with the probability

$$\frac{\binom{k/2}{p}^2\binom{n-k-(t-2p)}{l}}{\binom{n-k}{l}\binom{k}{2p}}.$$

In [32] a PARI/GP script is provided, which estimates the cost of the ISD attack in [31].

4 Proposal

We will present here a new variant of the McEliece cryptosystem using GRS codes, proposed by Bolkema, Gluesing-Luerssen, Kelley, Lauter, Malmskog and Rosenthal in [9]. Due to the distinguisher attacks [18, 14, 13], we want to change the scrambling matrices, such that the square code of the public matrix does not reveal any information of the secret code. To ensure this, we want the Schur matrix of the public matrix to have full rank.

The idea of the proposed cryptosystem is to use instead of a permutation matrix, an invertible matrix of row and column weight two. Note that $q \neq 2$ as there are no invertible matrices of row and column weight two over $\mathbb{F}_2$, since the all one vector lies in the kernel of such a row and column weight two over $\mathbb{F}_2$.

Please note that the proposed cryptosystem is a special case of the BBCRS scheme [4], by taking $m = 2, z = 0$ and hence $R = 0$.

We will state in the following the proposed cryptosystem in the McEliece and in the Niederreiter version.

4.1 McEliece Version

Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ integers. Let G be a $k \times n$ generator matrix of $\mathrm{GRS}_{n,k}(\alpha, \beta)$ code over $\mathbb{F}_q^n$, which is able to correct up to $\lfloor \frac{n-k}{2} \rfloor = t$ errors. We choose a $k \times k$ invertible matrix S, and a $n \times n$ invertible matrix Q, which is of row and column weight two, both over $\mathbb{F}_q$. We define $t_{\mathrm{pub}} = \lfloor \frac{t}{2} \rfloor$ and compute

$$G' = S^{-1}GQ^{-1}.$$

$$\begin{aligned} \text{Public Key} &= (G', t_{\mathrm{pub}}), \\ \text{Private Key} &= (G, S, Q). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_q^k$ and a random error vector $e \in \mathbb{F}_q^n$, s.t. $\mathrm{wt}(e) \leq t_{\mathrm{pub}}$ and compute the cipher

$$y = xG' + e.$$

Decryption: Compute

$$y' = yQ = xS^{-1}G + eQ.$$

Since $\mathrm{wt}(eQ) \leq t$ we can decode with decoding algorithm of of the $\mathrm{GRS}_{n,k}(\alpha, \beta)$ code and get xS^{-1}. By multiplication by S we get the message x.

The Niederreiter version is the dual of the McEliece version and has equivalent security.

4.2 Niederreiter Version

Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ integers. Let H be a $r \times n$ parity check matrix of $\mathrm{GRS}_{n,k}(\alpha, \beta)$ code over $\mathbb{F}_q$, with $r = n - k$. We choose a $r \times r$ invertible matrix S, and a $n \times n$ invertible matrix Q, which is of row and column weight two, both over $\mathbb{F}_q$. We define $t_{\mathrm{pub}} = \lfloor \frac{t}{2} \rfloor$ and compute

$$H' = S^{-1} H Q^T.$$

$$\begin{aligned} \text{Public Key} &= (H', t_{\mathrm{pub}}), \\ \text{Private Key} &= (H, S, Q). \end{aligned}$$

Encryption: Choose a message $x \in \mathbb{F}_q^n$, s.t. $\mathrm{wt}(x) \leq t_{\mathrm{pub}}$ and compute the cipher

$$y = H' x^T.$$

Decryption: Compute

$$y' = Sy = HQ^T x^T.$$

Since $\mathrm{wt}(Q^T x^T) \leq t$ we can do syndrome decoding and get $Q^T x^T$. By multiplication of the inverse of Q^T we get the message x.

5 Security

In this chapter we will show security of the proposed cryptosystem in [9] against the following selected attacks: The attack of Sidelnikov and Shestakov [37], the distinguisher attack [18, 13], the extended distinguisher attack [14] and the ISD attack [31].

5.1 Sidelnikov and Shestakov

This attack recovers for the public matrix SHP the matrix HP, where S is an invertible matrix and HP is a generator matrix of a GRS code, which is permutation equivalent to the secret GRS code. To know HP is enough to decrypt. Hence this attacks takes away the influence of the invertible matrix, for a public code which is permutation equivalent to the secret code. This attack is not applicable on the proposed cryptosystem, since the public code is not permutation equivalent to the secret GRS code.

5.2 Distinguisher Attack

The attack of Couvreur, Gaborit, Gauthier-Umaña, Otmani and Tillich is an attack on the BBCRS scheme [4], which the proposed cryptosystem is a special case of.

To avoid this attack is the main part of this thesis.

The attack uses that the public code of the BBCRS scheme has a low square code dimension. The attack reveals a large subcode of the public code, which has as square code a GRS code. With this code one finds a GRS code which is permutation equivalent to the secret code. To avoid the distinguisher attack we want to show that in the proposed cryptosystem the square code of the public code has maximal dimension. Since then an attacker can not distinguish if a GRS code or a random code was used as secret code.

5.2.1 Idea of the Argument

Observe that we will not consider the invertible matrix S, since multiplication by S results in a linear combination of the rows, which generates the same code.

The generator matrix, resp. the parity check matrix of the GRS code is fixed for the proposed cryptosystem. Unfortunately not every invertible matrix of row and column weight two is such that if multiplied to the generator matrix of a GRS code, the square code of the product has maximal dimension.
We include here an example computed with Sage [34].

Example 2. In the Niederreiter version, for the parameters $q = 5, n = 3, r = 2$. The public matrix is given by HQ^T, where H is a generator matrix of a $\mathrm{GRS}_{n,r}(\alpha, \beta)$ code. If $\alpha = (1, 2, 4)$ and $\beta = (4, 3, 3)$, then H is of the following form:

$$H = \begin{bmatrix} 4 & 3 & 3 \\ 4 & 1 & 2 \end{bmatrix}.$$

If we choose Q^T, the invertible $n \times n$ matrix of row and column weight two to

be

$$Q^T = \begin{bmatrix} 1 & 0 & 4 \\ 1 & 1 & 0 \\ 0 & 2 & 1 \end{bmatrix},$$

then the Schur matrix of HQ^T is the following matrix:

$$S(HQ^T) = \begin{bmatrix} 4 & 1 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 4 \end{bmatrix}.$$

And we can observe that $\det(S(HQ^T)) = 0$. $S(HQ^T)$ is thus not of full rank and hence the square code of the public matrix is not of maximal dimension.

The argument, why the proposed cryptosystem is secure against the distinguisher attack is in proving the following two points:

- For each generator matrix of a GRS code there exists an invertible matrix of row and column weight two, s.t. the public matrix will have maximal square code dimension.
- For each generator matrix of a GRS code the probability of a random invertible matrix of row and column weight two, to satisfy that the public matrix is of maximal square code dimension tends to one for $q \to \infty$.

In this argument we want to fix a special form of the row and column weight two matrices.

Remark 4. Note that every $n \times n$ matrix R over $\mathbb{F}_q$ of row and column weight two can be written as

$$R = PX + P'Y,$$

where P, P' are two $n \times n$ permutation matrices of disjoint support, corresponding to permutations σ, σ'. X is the diagonal matrix of $x = (x_1, \ldots, x_n) \in (\mathbb{F}_q^\times)^n$ and Y is respectively the diagonal matrix of $y = (y_1, \ldots, y_n) \in (\mathbb{F}_q^\times)^n$.

Let Q_n be a $n \times n$ matrix of row and column weight two of the following form

$$Q_n = \begin{bmatrix} x_1 & & & y_n \\ y_1 & x_2 & & \\ & \ddots & \ddots & \\ & & y_{n-1} & x_n \end{bmatrix}, \tag{10}$$

i.e.

$$Q = PX + P'Y,$$

with $\sigma = \mathrm{id}, \sigma' = (n \cdots 1)$.

Unfortunately not every row and column weight two matrix is permutation equivalent to a matrix of the form (10), but to a matrix, which has in the diagonal block matrices of this form.

Remark 5. For every $n \times n$ matrix R over $\mathbb{F}_q$ of row and column weight two, there exist permutation matrices P, P', s.t.

$$PRP' = \begin{bmatrix} Q^1_{n_1} & & \\ & \ddots & \\ & & Q^l_{n_l} \end{bmatrix} \tag{11}$$

where $Q^i_{n_i}$ are $n_i \times n_i$ matrices of the form (10) for $1 \leq i \leq l$ and $1 \leq l < n$.

The idea of the algorithm, which proves this remark, is to go through a row after a column j and take the first nonzero entry and set the entire column to be the jth column. Then we go through the jth column after the $j+1$th row and take the first nonzero entry and set the entire row as $j+1$th row. We provide here the pseudo code of this algorithm.

Input: R a row and column weight 2 matrix with the entries $a_{i,j}$. We denote the rows by r_i and the columns by c_j for $1 \leq i, j \leq n$.
Output: PRP' of the form (11).
Algorithm:

```
i ← 1
while i ≤ n do
    take l ≥ i the smallest integer, s.t. a_{i,l} ≠ 0
    switch c_l and c_i
    if exists k > i, s.t. a_{k,i} ≠ 0 and a_{k',i} = 0 ∀ i < k' < k then
        switch r_{i+1} and r_k
        i ← i + 1
    else
        i ← i + 1
    end if
end while
```

Hence in each column j of PRP' we have the nonzero entries x_j, y_j. We have chosen the form of the row and column weight two matrices, s.t. we can write down the determinant and the adjoint explicitly.

We have the following properties of Q in the considered form (10).
The determinant of Q is given by the following lemma.

Lemma 7. Let $\mathbb{F}_q$ be a finite field and $2 \leq n \leq q$ be an integer. Let Q be of the form (10), then

$$\det(Q) = \prod_{i=1}^{n} x_i + (-1)^{n+1} \prod_{i=1}^{n} y_i.$$

Proof. We compute the determinant of Q by Laplace expansion in the nth column. We denote by $M_{i,j}$ the $(n-1) \times (n-1)$ submatrix of Q by deleting the ith row and the jth column, hence

$$M_{1,n} = \begin{pmatrix} y_1 & x_2 & & \\ & \ddots & \ddots & \\ & & y_{n-2} & x_{n-1} \\ & & & y_{n-1} \end{pmatrix}$$

is an upper triangular matrix and

$$M_{n,n} = \begin{pmatrix} x_1 & & & \\ y_1 & x_2 & & \\ & \ddots & \ddots & \\ & & y_{n-2} & x_{n-1} \end{pmatrix}$$

is a lower triangular matrix. Hence we have the following for their determinants.

$$\begin{aligned} \det(M_{1,n}) &= y_1 \cdots y_{n-1}, \\ \det(M_{n,n}) &= x_1 \cdots x_{n-1}. \end{aligned}$$

Hence

$$\begin{aligned} \det(Q) &= (-1)^{n+1} y_n \det(M_{1,n}) + (-1)^{n+n} \det(M_{n,n}) \\ &= (-1)^{n+1} \prod_{i=1}^{n} y_i + \prod_{i=1}^{n} x_i. \end{aligned}$$

□

The adjoint of Q, which is the transpose of the cofactor matrix, is given by the following lemma.

Lemma 8. Let $\mathbb{F}_q$ be a finite field and $2 \leq n \leq q$ be an integer. Let Q be of the form (10), then the adjoint of Q denoted by $\widetilde{Q}$ can be written as

$$\widetilde{Q} = (\tilde{q}_{i,j})_{1 \leq i,j \leq n}, \tag{12}$$

with

i)
$$\tilde{q}_{i,i} = \prod_{\substack{k=1 \\ k \neq i}}^{n} x_k,$$

ii) for $j < i$
$$\tilde{q}_{i,j} = (-1)^{i+j} \prod_{l=1}^{j-1} x_l \prod_{k=j}^{i-1} y_k \prod_{m=i+1}^{n} x_m,$$

iii) for $j > i$
$$\tilde{q}_{i,j} = (-1)^{i+j} (-1)^n \prod_{k=1}^{i-1} y_k \prod_{l=i+1}^{j-1} x_l \prod_{m=j}^{n} y_m.$$

Hence every entry of $\widetilde{Q}$ is a polynomial in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$ of total degree $n-1$ and in $n-1$ variables.

Proof. Let $\bar{Q}$ denote the cofactor matrix of Q, i.e. $\widetilde{Q}^T = \bar{Q}$. $\bar{Q}$ has entries $\bar{q}_{i,j} = (-1)^{i+j} \det(M_{i,j})$, where $M_{i,j}$ is the $(n-1) \times (n-1)$ submatrix of Q by deleting the ith row and jth column.

i) Hence

$$\bar{q}_{i,i} = (-1)^{i+i}\det(M_{i,i}).$$

Observe that $M_{i,i}$ has the following form:

$$M_{i,i} = \begin{pmatrix} x_1 & & & & & & & & y_n \\ y_1 & x_2 & & & & & & & \\ & \ddots & \ddots & & & & & & \\ & & y_{i-2} & x_{i-1} & & & & & \\ & & & 0 & x_{i+1} & & & & \\ & & & & y_{i+1} & x_{i+2} & & & \\ & & & & & \ddots & \ddots & & \\ & & & & & & y_{n-1} & x_n \end{pmatrix}.$$

We compute its determinant as in Lemma 7, hence

$$\det(M_{i,i}) = \prod_{\substack{k=1 \\ k\neq i}}^{n} x_k.$$

ii) Let $j < i$, hence in the cofactor matrix $\bar{Q}$ we have $i < j$ by switching j and i. If $i = n$, thus in the cofactor matrix we have $j = n$, we observe that $M_{i,n}$ has the following form.

$$M_{i,n} = \begin{pmatrix} x_1 & & & & & & & & \\ y_1 & \ddots & & & & & & & \\ & \ddots & x_{i-2} & & & & & & \\ & & y_{i-2} & x_{i-1} & & & & & \\ & & & 0 & y_i & x_{i+1} & & & \\ & & & & & \ddots & \ddots & & \\ & & & & & & y_{n-2} & x_{n-1} & \\ & & & & & & & y_{n-1} & \end{pmatrix}.$$

Thus the matrix $M_{i,n}$ can be written as

$$M_{i,n} = \begin{pmatrix} A & \mathbf{0} \\ \mathbf{0} & B \end{pmatrix},$$

where $\mathbf{0}$ denotes the zero matrix. Hence

$$\det(M_{i,n}) = \det(A)\det(B)$$

The matrix A is a lower triangular matrix, having the determinant

$$\prod_{k=1}^{i-1} x_k.$$

And the matrix B is an upper triangular matrix having the determinant

$$\prod_{k=i}^{n-1} y_k.$$

By switching i and j we get the claim for $i = n$.
If $i \neq n$, then we observe that $M_{i,j} = \bar{M}$ has the following form.

$$\bar{M} = \begin{bmatrix}
x_1 & & & & & & & & & & & & y_n \\
y_1 & \ddots & & & & & & & & & & & \\
 & \ddots & x_{i-2} & & & & & & & & & & \\
 & & y_{i-2} & x_{i-1} & & & & & & & & & \\
 & & & 0 & y_i & x_{i+1} & & & & & & & \\
 & & & & & \ddots & \ddots & & & & & & \\
 & & & & & & y_j & x_{j-1} & & & & & \\
 & & & & & & & y_{j-1} & 0 & & & & \\
 & & & & & & & 0 & x_{j+1} & & & & \\
 & & & & & & & & y_{j+1} & x_{j+2} & & & \\
 & & & & & & & & & \ddots & \ddots & & \\
 & & & & & & & & & & y_{n-1} & x_n
\end{bmatrix}.$$

We compute the determinant of $\bar{M}$ by Laplace expansion in the last column, hence

$$\det(\bar{M}) = (-1)^n y_n \det(\bar{M}_{1,n-1}) + x_n \det(\bar{M}_{n-1,n-1}).$$

Where $\bar{M}_{1,n-1}$ is given by the matrix $\bar{M}$ deleting the first row and the last column, this makes the matrix upper triangular and since it has zeros on its diagonal, the determinant of $\bar{M}_{1,n-1}$ is zero.

$$\bar{M}_{1,n-1} = \begin{bmatrix}
y_1 & x_2 & & & & & & & & & & \\
 & \ddots & \ddots & & & & & & & & & \\
 & & y_{i-2} & x_{i-1} & & & & & & & & \\
 & & & 0 & y_i & x_{i+1} & & & & & & \\
 & & & & & \ddots & \ddots & & & & & \\
 & & & & & & y_j & x_{j-1} & & & & \\
 & & & & & & & y_{j-1} & 0 & & & \\
 & & & & & & & 0 & x_{j+1} & & & \\
 & & & & & & & & y_{j+1} & \ddots & & \\
 & & & & & & & & & \ddots & x_{n-1} & \\
 & & & & & & & & & & y_{n-1} &
\end{bmatrix}.$$

Whereas the matrix $\bar{M}_{n-1,n-1}$ is given by the following matrix

$$\bar{M}_{n-1,n-1} = \begin{bmatrix} x_1 & & & & & & & & & & & \\ y_1 & \ddots & & & & & & & & & & \\ & \ddots & x_{i-2} & & & & & & & & & \\ & & y_{i-2} & x_{i-1} & & & & & & & & \\ & & & 0 & y_i & x_{i+1} & & & & & & \\ & & & & & \ddots & \ddots & & & & & \\ & & & & & & y_j & x_{j-1} & & & & \\ & & & & & & & y_{j-1} & 0 & & & \\ & & & & & & & 0 & x_{j+1} & & & \\ & & & & & & & & y_{j+1} & x_{j+2} & & \\ & & & & & & & & & \ddots & \ddots & \\ & & & & & & & & & & y_{n-2} & x_{n-1} \end{bmatrix}.$$

Thus the matrix $\bar{M}_{n-1,n-1}$ can be written as

$$\bar{M}_{n-1,n-1} = \begin{pmatrix} A & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & B & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & C \end{pmatrix},$$

where $\mathbf{0}$ denotes the zero matrix. Hence

$$\det(\bar{M}_{n-1,n-1}) = \det(A)\det(B)\det(C).$$

The matrix A is a lower triangular matrix, having the determinant

$$\prod_{k=1}^{i-1} x_k.$$

And the matrix B is an upper triangular matrix having the determinant

$$\prod_{k=i}^{j-1} y_k.$$

The matrix C is a lower triangular matrix, having the determinant

$$\prod_{k=j+1}^{n-1} x_k.$$

And therefore we have that

$$\det(M_{i,j}) = x_n \det(\bar{M}_{n-1,n-1}) = \prod_{l=1}^{i-1} x_l \prod_{k=i}^{j-1} y_k \prod_{m=j+1}^{n} x_m.$$

To get $\widetilde{Q}$ we have to switch i and j.

iii) Let $j > i$, hence in the cofactor matrix $\bar{Q}$ we have $i > j$ by switching j and i. We observe that $M_{i,j} = \bar{M}$ has the following form.

$$\bar{M} = \begin{bmatrix} x_1 & & & & & & & & & & & & y_n \\ y_1 & \ddots & & & & & & & & & & & \\ & \ddots & x_{j-1} & & & & & & & & & & \\ & & y_{j-1} & 0 & & & & & & & & & \\ & & & x_{j+1} & & & & & & & & & \\ & & & y_{j+1} & \ddots & & & & & & & & \\ & & & & \ddots & x_{i-2} & & & & & & & \\ & & & & & y_{i-2} & x_{i-1} & & & & & & \\ & & & & & & 0 & y_i & x_{i+1} & & & & \\ & & & & & & & & y_{i+1} & x_{i+2} & & & \\ & & & & & & & & & \ddots & \ddots & & \\ & & & & & & & & & & y_{n-1} & x_n \end{bmatrix}.$$

We compute the determinant of $\bar{M}$ by Laplace expansion in the last column, hence

$$\det(\bar{M}) = (-1)^n y_n \det(\bar{M}_{1,n-1}) + x_n \det(\bar{M}_{n-1,n-1}).$$

Where $\bar{M}_{n-1,n-1}$ is given by the matrix $\bar{M}$ deleting the last row and the last column, this makes the matrix lower triangular and since it has zeros on its diagonal, the determinant of $\bar{M}_{n-1,n-1}$ is zero.

$$\bar{M}_{n-1,n-1} = \begin{bmatrix} x_1 & & & & & & & & & & & \\ y_1 & \ddots & & & & & & & & & & \\ & \ddots & x_{j-1} & & & & & & & & & \\ & & y_{j-1} & 0 & & & & & & & & \\ & & & x_{j+1} & & & & & & & & \\ & & & y_{j+1} & \ddots & & & & & & & \\ & & & & \ddots & x_{i-2} & & & & & & \\ & & & & & y_{i-2} & x_{i-1} & & & & & \\ & & & & & & 0 & y_i & x_{i+1} & & & \\ & & & & & & & & y_{i+1} & x_{i+2} & & \\ & & & & & & & & & \ddots & \ddots & \\ & & & & & & & & & & y_{n-2} & x_{n-1} \end{bmatrix}.$$

Whereas the matrix $\bar{M}_{1,n-1}$ is given by the following matrix

$$\bar{M}_{1,n-1} = \begin{bmatrix} y_1 & x_2 & & & & & & & & & & \\ & \ddots & \ddots & & & & & & & & & \\ & & y_{j-2} & x_{j-1} & & & & & & & & \\ & & & y_{j-1} & 0 & & & & & & & \\ & & & & x_{j+1} & & & & & & & \\ & & & & y_{j+1} & \ddots & & & & & & \\ & & & & & \ddots & x_{i-2} & & & & & \\ & & & & & & y_{i-2} & x_{i-1} & & & & \\ & & & & & & & 0 & y_i & x_{i+1} & & \\ & & & & & & & & & y_{i+1} & \ddots & \\ & & & & & & & & & & \ddots & x_{n-1} \\ & & & & & & & & & & & y_{n-1} \end{bmatrix}.$$

Thus the matrix $\bar{M}_{1,n-1}$ can be written as

$$\bar{M}_{n-1,n-1} = \begin{pmatrix} A & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & B & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & C \end{pmatrix},$$

where $\mathbf{0}$ denotes the zero matrix. Hence

$$\det(\bar{M}_{1,n-1}) = \det(A)\det(B)\det(C).$$

The matrix A is an upper triangular matrix, having the determinant

$$\prod_{k=1}^{j-1} y_k.$$

The matrix B is a lower triangular matrix having the determinant

$$\prod_{k=j+1}^{i-1} x_k.$$

The matrix C is an upper triangular matrix, having the determinant

$$\prod_{k=i}^{n-1} y_k.$$

And therefore we have that

$$\det(M_{i,j}) = (-1)^n y_n \det(\bar{M}_{1,n-1}) = (-1)^n \prod_{k=1}^{j-1} y_k \prod_{l=j+1}^{i-1} x_l \prod_{m=i}^{n} y_m.$$

To get $\widetilde{Q}$ we have to switch i and j.

□

Theorem 3. Let $\mathbb{F}_q$ be a finite field. Let $2 \leq n \leq q$ be an integer. Let Q be of the form (10), s.t. Q is invertible. Then

$$Q^{-1} = \frac{1}{\det(Q)}\widetilde{Q} = (r_{i,j})_{1\leq i,j\leq n},$$

with

i)

$$r_{i,i} = \frac{1}{\prod_{i=1}^{n} x_i + (-1)^{n+1}\prod_{i=1}^{n} y_i} \prod_{\substack{k=1 \\ k\neq i}}^{n} x_k,$$

ii) for $j < i$

$$r_{i,j} = \frac{1}{\prod_{i=1}^{n} x_i + (-1)^{n+1}\prod_{i=1}^{n} y_i}(-1)^{i+j} \prod_{l=1}^{j-1} x_l \prod_{k=j}^{i-1} y_k \prod_{m=i+1}^{n} x_m,$$

iii) for $j > i$

$$r_{i,j} = \frac{1}{\prod_{i=1}^{n} x_i + (-1)^{n+1}\prod_{i=1}^{n} y_i}(-1)^{i+j}(-1)^{n} \prod_{k=1}^{i-1} y_k \prod_{l=i+1}^{j-1} x_l \prod_{m=j}^{n} y_m.$$

Proof. By Lemma 7 and Lemma 8 we have the formula for the determinant of Q and the formula for the adjoint of Q, denoted by $\widetilde{Q}$. We get the claim since $Q^{-1} = \frac{1}{\det(Q)}\widetilde{Q}$. □

Hence for R of the form (11), i.e.

$$\begin{bmatrix} Q^1_{n_1} & & \\ & \ddots & \\ & & Q^l_{n_l} \end{bmatrix},$$

we have that the adjoint of R is of the following form

$$\widetilde{R} = \begin{bmatrix} \widetilde{Q}^1_{n_1} & & \\ & \ddots & \\ & & \widetilde{Q}^l_{n_l} \end{bmatrix},$$

where $\widetilde{Q}^i_{n_i}$ denotes the adjoint of $Q^i_{n_i}$.

And the determinant of R can be computed as

$$\det(R) = \prod_{i=1}^{l} \det(Q^i_{n_i}).$$

Please observe that $\widetilde{R}_n$ has in each column all $2n$ variables of R_n but two and each entry is a polynomial in $n-1$ variables. This is also true for the inverse of R_n.

We want to state some remarks on parameters we do not consider.

- If $r = 1$ in the Niederreiter version, resp. $k = 1$ in the McEliece version it is easy to see, that for all invertible row and column weight two matrices R_n the Schur matrix of the public matrix will have maximal rank 1, hence we will assume $r > 1$, resp. $k > 1$.
- We have excluded $q = 2$ as there exist no invertible matrices of row and column weight two over $\mathbb{F}_2$. Since the all one vector lies in the kernel of a row and column weight two matrix over $\mathbb{F}_2$.
- The following results will not hold for $q = 3$, which implies $n = 3, r = 2$ in the Niederreiter version, resp. $k = 2$ in the McEliece version. One can compute by Sage [34] that $\frac{1}{4}$ of all invertible matrices of row and column weight two are s.t. the Schur matrix of the public matrix will have maximal rank. Hence we can assume from now on $q > 3$.

5.2.2 Niederreiter Version

We will need the following notations.

In the Niederreiter version the public matrix is given by $H_{n,r}R_n^T$.
Where $H_{n,r}$ is a generator matrix of a $\mathrm{GRS}_{n,r}(\alpha, \beta)$ code, with $r = n - k$, thus we can write $H_{n,r}$ as

$$H_{n,r} = \begin{bmatrix} \beta_1 & \cdots & \beta_n \\ \beta_1\alpha_1 & \cdots & \beta_n\alpha_n \\ \vdots & & \vdots \\ \beta_1\alpha_1^{r-1} & \cdots & \beta_n\alpha_n^{r-1} \end{bmatrix}, \tag{13}$$

with $\alpha = (\alpha_1, \ldots, \alpha_n)$ s.t. $\alpha_i \neq \alpha_j \ \forall i, j \in \{1, \ldots, n\}$ with $i \neq j$ and $\beta = (\beta_1, \ldots, \beta_n)$ s.t. $\beta_i \neq 0 \ \forall i \in \{1, \ldots, n\}$.

Let R_n^T be of the form (11), with $x_i, y_i \in \mathbb{F}_q^\times$ for $1 \leq i \leq n$ s.t. $\det(R_n) \neq 0$.
Whereas $H_{n,r}$ is considered to be fixed, we want to examine the choice of the matrices R_n of row and column weight two.

We recall that for A a $r \times n$ matrix, we denote by $S(A)$ the Schur matrix of A, which is of size $\frac{1}{2}(r^2 + r) \times n$, hence the maximal rank of $S(A)$ is by Proposition 6

$$m = \min\left\{n, \frac{1}{2}(r^2 + r)\right\}. \tag{14}$$

Define

$$\begin{aligned} A_n &= \{R_n \in \mathrm{GL}_n(\mathbb{F}_q) \mid R_n \text{ is of the form (11)}\}, \\ \mathcal{G}_{H_{n,r}} &= \left\{R_n^T \in A_n \mid S(H_{n,r}R_n^T) \text{ is of full rank } m\right\}. \end{aligned}$$

Please observe that in our computations we do the following. We take a random row and column weight two matrix R, we find the permutation matrices P and P', s.t. PRP' is of the form (11). This form is now fixed and we want

to examine the amount of choices for the nonzero entries of the matrix.

We want to compute the size of A_n, which is the total amount of invertible $n \times n$ matrices of the form (11). Since these matrices are defined by their $2n$ nonzero entries, we can fix all of them but one, and choose the last variable s.t. the determinant which is linear in this variable, is nonzero. Thus we get $\mid A_n \mid = (q-1)^{2n} - (q-1)^{2n-1}$. We observe that the probability for a random $R_n \in A_n$ to be in $\mathcal{G}_{H_{n,r}}$ can be computed as

$$\begin{aligned} \frac{\mid \mathcal{G}_{H_{n,r}} \mid}{\mid A_n \mid} &= \frac{\mid \mathcal{G}_{H_{n,r}} \mid}{(q-1)^{2n} - (q-1)^{2n-1}} \\ &= \frac{\mid \mathcal{G}_{H_{n,r}} \mid}{(q-1)^{2n}} \left(\frac{q-1}{q-2}\right) \geq \frac{\mid \mathcal{G}_{H_{n,r}} \mid}{(q-1)^{2n}}. \end{aligned}$$

Thus it is enough to consider

$$\frac{\mid \mathcal{G}_{H_{n,r}} \mid}{(q-1)^{2n}}$$

to give a lower bound on the probability, that R_n is in $\mathcal{G}_{H_{n,r}}$. Therefore we set the nonzero entries of R_n as variables.

For fixed $1 \leq r < n \leq q$ we have a fixed $H_{n,r}$ as in (13) and want to compute a lower bound on the size of $\mathcal{G}_{H_{n,r}}$.

1. case: $n \leq \frac{1}{2}(r^2 + r)$

Please observe that in practice, we mostly take $r = \lfloor \frac{n}{2} \rfloor$, hence this case is the most common one in practice. We state in the following a lower bound on the size of $\mathcal{G}_{H_{n,r}}$.

Theorem 4. Let $\mathbb{F}_q$ be a finite field and $1 \leq r < n \leq q$ be integers, s.t. $n \leq \frac{1}{2}(r^2 + r)$. Under the assumption that there exists a nontrivial minor of $S(H_{n,r}R_n^T)$, then we have the following lower bound on the size of $\mathcal{G}_{H_{n,r}}$.

$$\mid \mathcal{G}_{H_{n,r}} \mid \geq ((q-1)^2 - 2(q-1))^n.$$

With this we can state the following two corollaries, about the existence of R_n in $\mathcal{G}_{H_{n,r}}$ and the probability that a random $R_n \in A_n$ is in $\mathcal{G}_{H_{n,r}}$.

Corollary 1. We have the existence of R_n in $\mathcal{G}_{H_{n,r}}$. Since $\mid \mathcal{G}_{H_{n,r}} \mid \geq 1$ for $q > 3$.

Corollary 2. The probability of $R_n \in A_n$ to be in $\mathcal{G}_{H_{n,r}}$ is greater than or equal to

$$\frac{((q-1)^2 - 2(q-1))^n}{(q-1)^{2n}} = \left(1 - \frac{2}{q-1}\right)^n.$$

And we can observe that for fixed n this quantity tends to one for $q \to \infty$.

For the proof of the theorem we need the following lemma.

Lemma 9. Let $\mathbb{F}_q$ be a finite field and $1 \leq n \leq q$ integers. Let p be a nontrivial homogeneous polynomial in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$, of total degree $2n$, in each variable of degree at most 2, which has that each monomial is of the form

$$\prod_{i=1}^{n} x_i^{d_i} y_i^{2-d_i},$$

for $0 \leq d_i \leq 2, \ \forall\ 1 \leq i \leq n$. Then there exist at least

$$((q-1)^2 - 2(q-1))^n$$

choices for the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$ in $\mathbb{F}_q^\times$, s.t. p evaluated in these choices is nonzero.

Proof. We prove this by induction over n.

For $n = 1$ we can write

$$p(x_1, y_1) = x_1^2 a + x_1 y_1 b + y_1^2 c,$$

with $a, b, c \in \mathbb{F}_q$ and p is nontrivial in $\mathbb{F}_q[x_1, y_1]$. We can fix y_1 to be in $\mathbb{F}_q^\times$ and get that we have for x_1 at most two roots. Thus we have to take away from the total amount of possibilities for x_1, y_1 in $\mathbb{F}_q^\times$ for each choice of y_1 two choices of x_1. Hence we get $((q-1)^2 - 2(q-1))$.

The induction hypothesis states that for a polynomial p, which is homogeneous and nontrivial in $\mathbb{F}_q[x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}]$, of total degree $2(n-1)$, in each variable of degree at most 2 and has that each monomial is of the form

$$\prod_{i=1}^{n-1} x_i^{d_i} y_i^{2-d_i},$$

for $0 \leq d_i \leq 2, \ \forall\ 1 \leq i \leq n-1$, there exist at least

$$((q-1)^2 - 2(q-1))^{n-1}$$

choices for the variables $x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}$ in $\mathbb{F}_q^\times$, s.t. p evaluated in these choices is nonzero.

Let us assume we have p as in the lemma. We observe that we can write p as

$$x_n^2 a + x_n y_n b + y_n^2 c.$$

with $a, b, c \in \mathbb{F}_q[x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}]$. By the assumption we have that at least one of the polynomials a, b, c is nontrivial. Let us assume w.l.o.g. that a is nontrivial and we observe that a satisfies all conditions of the induction hypothesis. Thus we have

$$((q-1)^2 - 2(q-1))^{n-1}$$

choices of the variables $x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}$ in $\mathbb{F}_q^\times$, s.t. a evaluated in these choices is nonzero. Let us fix one of these choices and define $\bar{a}$ to be the

evaluation of a in this choice and define $\bar{b}$ and $\bar{c}$ respectively. With this we can write p as

$$x_n^2\bar{a} + x_n y_n \bar{b} + y_n^2 \bar{c},$$

with $\bar{a} \in \mathbb{F}_q^\times$ and p is by assumption nontrivial. Thus by the case of $n = 1$, we have $((q-1)^2 - 2(q-1))$ choices for x_n, y_n in $\mathbb{F}_q^\times$ s.t. p evaluated in these choices is nonzero. Hence we get in total

$$((q-1)^2 - 2(q-1))^{n-1}((q-1)^2 - 2(q-1)) = ((q-1)^2 - 2(q-1))^n.$$

□

Now we can prove Theorem 4.

Proof. By the assumption there exists a nontrivial minor of $S(H_{n,r}R_n^T)$ in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$. Let us fix a nontrivial minor of $S(H_{n,r}R_n^T)$, by choosing n rows. Observe that this minor is a polynomial in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$, which satisfies the properties of Lemma 9. Hence there exist

$$((q-1)^2 - 2(q-1))^n$$

choices of the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$ in $\mathbb{F}_q^\times$, s.t. the minor evaluated in these choices is nonzero. Hence we get the claim. □

2. case: $n \geq \frac{1}{2}(r^2 + r)$

We state in the following a lower bound on the size of $\mathcal{G}_{H_{n,r}}$.

Theorem 5. Let $\mathbb{F}_q$ be a finite field and $1 \leq r < n \leq q$ be integers, s.t. $n \geq \frac{1}{2}(r^2 + r) = m$. Under the assumption that there exists a nontrivial minor of $S(H_{n,r}R_n^T)$, we have the following lower bound on the size of $\mathcal{G}_{H_{n,r}}$.

$$\left| \mathcal{G}_{H_{n,r}} \right| \geq ((q-1)^2 - 2(q-1))^m (q-1)^{2(n-m)}.$$

With this we can state the following two corollaries, about the existence of R_n in $\mathcal{G}_{H_{n,r}}$ and the probability that a random $R_n \in A_n$ is in $\mathcal{G}_{H_{n,r}}$.

Corollary 3. We have the existence of R_n in $\mathcal{G}_{H_{n,r}}$. Since $\left| \mathcal{G}_{H_{n,r}} \right| \geq 1$ for $q > 3$.

Corollary 4. The probability of $R_n \in A_n$ to be in $\mathcal{G}_{H_{n,r}}$ is greater than or equal to

$$\frac{((q-1)^2 - 2(q-1))^m (q-1)^{2(n-m)}}{(q-1)^{2n}} = \left(1 - \frac{2}{q-1}\right)^m,$$

where $m = \frac{1}{2}(r^2 + r)$. And we can observe that for fixed n, r this quantity tends to one for $q \to \infty$.

Proof. By assumption there exists a nontrivial minor of $S(H_{n,r}R_n^T)$. Let us fix a nontrivial minor of $S(H_{n,r}R_n^T)$, by choosing m columns. Observe that this minor is a polynomial in $\mathbb{F}_q[x_1, \ldots, x_m, y_1, \ldots, y_m]$, which satisfies the properties of Lemma 9. Hence there exist

$$((q-1)^2 - 2(q-1))^m$$

choices of the variables $x_1, \ldots, x_m, y_1, \ldots, y_m$ in $\mathbb{F}_q^\times$, s.t. the minor evaluated in these choices is nonzero. For the rest of the $2(n-m)$ variables we have only the condition to be in $\mathbb{F}_q^\times$. Hence we get the claim. □

Comparing the different cases we can summarize in the Niederreiter version that the probability of $R_n \in A_n$ to be in $\mathcal{G}_{H_{n,r}}$ is greater than or equal to

$$\left(1 - \frac{2}{q-1}\right)^m.$$

With m as in (14). We can observe that for fixed n, r this quantity tends to one for $q \to \infty$.

Thus we have achieved to show in the Niederreiter version, under the assumption of the existence of a nontrivial minor, the existence of $R_n \in \mathcal{G}_{H_{n,r}}$ for each $1 \leq r < n \leq q$ and each $H_{n,r}$ and that for large q, the probability that $R_n \in A_n$ is in $\mathcal{G}_{H_{n,r}}$ tends to one.

To make the proposed cryptosystem in the Niederreiter version secure against the distinguisher attack, one can test if the Schur matrix of the public matrix has full rank and then use this matrix Q to scramble the secure code. The complexity of this can be found in the Chapter 7.

Please observe that one could have also used this theorem.

Theorem 6. [Lemma of Schwartz-Zippel][[35], Corollary 1] Let $\mathbb{F}_q$ be a finite field and $\mathbb{F}_{q^n}$ be an extension field. Let $p \in \mathbb{F}_q[x_1, \ldots, x_n]$ be a nontrivial polynomial of total degree $d \geq 0$. Let S be a finite subset of $\mathbb{F}_{q^n}$. Let $a_1, \ldots, a_n$ be chosen randomly, uniformly and independent in S. Then

$$\mathbb{P}[p(a_1, \ldots, a_n) = 0] \leq \frac{d}{\mid S \mid}.$$

If we apply this to the case $n = \frac{1}{2}(r^2 + r)$, we set $p = \det(S(H_{n,r}R_n^T)) \in \mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$ of total degree $d = 2n$ and $S = \mathbb{F}_q^\times$. Then Theorem 6 gives the following result.

$$\mathbb{P}[p(a_1, \ldots, a_n, b_1, \ldots, b_n) = 0] \leq \frac{2n}{q-1}.$$

Thus this argument also shows, that for fixed n the probability of $R_n \in A_n$ to be in $\mathcal{G}_{H_{n,r}}$ tends to one for $q \to \infty$.

One has existence of R_n in $\mathcal{G}_{H_{n,r}}$, whenever

$$\frac{2n}{q-1} < 1 \Rightarrow 2n < q - 1.$$

Whereas with our argument we have existence of R_n in $\mathcal{G}_{H_{n,r}}$ for any $n \leq q$.

Comparing the probability of $R_n \in A_n$ to be in $\mathcal{G}_{H_{n,r}}$ achieved with our argument, and the probability achieved by using the lemma of Schwartz-Zippel, we note that we have a stronger argument. Since for $n \geq 2$

$$\left(1 - \frac{2}{q-1}\right)^n > 1 - \frac{2n}{q-1}.$$

The reason for this, is that the lemma of Schwartz-Zippel applies to any polynomial and does not use the fact, that p has in in each variable degree at most 2.

5.2.3 McEliece Version

In the McEliece version the public matrix is given by GQ^{-1}, where G is a generator matrix of a $\mathrm{GRS}_{n,k}(\alpha, \beta)$ code and Q is a matrix of row and column weight two. Whereas G is considered to be fixed, we want to examine the choice of the matrix Q, which we choose to be of the form (11), denoted by R_n.

We have the following property of the Schur matrix.

Remark 6. Let A be a $k \times n$ matrix over $\mathbb{F}_q$ and $\lambda \in \mathbb{F}_q$. Then $S(\lambda A) = \lambda^2 S(A)$.

Due to this property and the assumption that R_n is invertible, hence $\det(R_n) \neq 0$, it is enough to consider the adjoint of R_n as in (12), denoted by $\widetilde{R}_n$, instead of the inverse.

We need the following notations.

In the McEliece version the public matrix is given by $G_{n,k}\widetilde{R}_n$.
Where $G_{n,k}$ is a generator matrix of a $\mathrm{GRS}_{n,k}(\alpha, \beta)$ code, thus we can write $G_{n,k}$ as

$$G_{n,k} = \begin{bmatrix} \beta_1 & \cdots & \beta_n \\ \beta_1\alpha_1 & \cdots & \beta_n\alpha_n \\ \vdots & & \vdots \\ \beta_1\alpha_1^{k-1} & \cdots & \beta_n\alpha_n^{k-1} \end{bmatrix}, \tag{15}$$

with $\alpha = (\alpha_1, \ldots, \alpha_n)$ s.t. $\alpha_i \neq \alpha_j \ \forall i, j \in \{1, \ldots, n\}$ with $i \neq j$ and $\beta = (\beta_1, \ldots, \beta_n)$ s.t. $\beta_i \neq 0 \ \forall i \in \{1, \ldots, n\}$. R_n is an invertible $n \times n$ matrix of row and column weight two, which is of the form (11).

We define

$$\mathcal{G}_{G_{n,k}} \quad = \quad \left\{ R_n \in A_n \;\middle|\; S(G_{n,k}\widetilde{R}_n) \text{ has full rank } m \right\}.$$

By Proposition 6 we recall that the maximal rank is

$$m = \min\left\{ n, \frac{1}{2}(k^2 + k) \right\}.$$

For fixed $1 \leq k < n \leq q$ we have a fixed $G_{n,k}$ as in (15) and want to compute a lower bound on the size of $\mathcal{G}_{G_{n,k}}$.

We want to apply the same argument in the McEliece version, as we did in the Niederreiter version.

1. case: $n \leq \frac{1}{2}(k^2 + k)$

Please observe that in practice, we mostly take $k = \lfloor \frac{n}{2} \rfloor$, hence this case is the most common one in practice. We state in the following a lower bound on the size of $\mathcal{G}_{G_{n,k}}$, for $q > 2n - 2$.

Theorem 7. Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ be integers, s.t. $n \leq \frac{1}{2}(k^2 + k)$ and $q > 2n - 2$. Under the assumption, that there exists a

nontrivial minor of $S(G_{n,k}\widetilde{R}_n)$, we have the following lower bound on the size of $\mathcal{G}_{G_{n,k}}$.

$$\mid \mathcal{G}_{G_{n,k}} \mid \geq ((q-1)^2 - 2(n-1)(q-1))^n.$$

With this we can state the following two corollaries, about the existence of R_n in $\mathcal{G}_{G_{n,k}}$ and the probability that a random $R_n \in A_n$ is in $\mathcal{G}_{G_{n,k}}$.

Corollary 5. We have the existence of R_n in $\mathcal{G}_{G_{n,k}}$. Since $\mid \mathcal{G}_{G_{n,k}} \mid \geq 1$ for $q > 3$.

Corollary 6. The probability of $R_n \in A_n$ to be in $\mathcal{G}_{G_{n,k}}$ is greater than or equal to

$$\frac{((q-1)^2 - 2(n-1)(q-1))^n}{(q-1)^{2n}} = \left(1 - \frac{2(n-1)}{q-1}\right)^n.$$

And we can observe that for fixed n this quantity tends to one for $q \to \infty$.

For the proof of the theorem we need the following lemma.

Lemma 10. Let $\mathbb{F}_q$ be a finite field and $1 \leq m < n \leq q$ integers with $q > 2m$. Let p be a nontrivial homogeneous polynomial in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$, of total degree $2m(n-1)$, in each variable of degree at most $2m$. Then there exist at least

$$((q-1)^2 - 2m(q-1))^n$$

choices for the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$ in $\mathbb{F}_q^\times$, s.t. p evaluated in these choices is nonzero.

Proof. We prove this by induction over the number of variables n.

For $n = 1$ the polynomial is of total degree zero, and nontrivial hence $p(x_1, y_1) = c \in \mathbb{F}_q^\times$.

The induction hypothesis states that for a polynomial p, which is homogeneous and nontrivial in $\mathbb{F}_q[x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}]$, and has total degree $2m(n-2)$ and in each variable of degree at most $2m$, there exist at least

$$((q-1)^2 - 2m(q-1))^{n-1}$$

choices for the variables $x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}$ in $\mathbb{F}_q^\times$, s.t. p evaluated in these choices is nonzero.

Let us assume we have a polynomial p as in the lemma. Let us look at p as a polynomial in $\mathbb{F}_q[x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}][x_n, y_n]$. W.l.o.g. we can assume that there exists a monomial with x_n which has nonzero coefficient in $\mathbb{F}_q[x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}]$. The nonzero coefficient of the monomial with x_n is a nontrivial polynomial in $\mathbb{F}_q[x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}]$, and has total degree $2m(n-2)$ and in each variable degree at most $2m$. Hence by induction hypothesis, there exist at least

$$((q-1)^2 - 2m(q-1))^{n-1}$$

choices of the variables $x_1, \ldots, x_{n-1}, y_1, \ldots, y_{n-1}$ in $\mathbb{F}_q^\times$, s.t. the coefficient evaluated in these choices is nonzero. Let us fix one of these choices, now the

polynomial is nontrivial in $\mathbb{F}_q[x_n, y_n]$. Let y_n be any element of $\mathbb{F}_q^\times$, then we have at most $2m$ roots for x_n in $\mathbb{F}_q^\times$. With the choices for x_n, y_n we get in total

$$((q-1)^2 - 2m(q-1))^{n-1}((q-1)^2 - 2m(q-1)) = ((q-1)^2 - 2m(q-1))^n.$$

Please note that an exception for this is when $q \leq 2m$, in this case we could exclude all of the values, e.g. $x^{q-1} - 1$ annihilates all elements of $\mathbb{F}_q^\times$. Thus we want to assume here that q is large enough, more precisely, that $q > 2m$. □

Now we can prove Theorem 7.

Proof. By the assumption there exists a nontrivial minor of $S(G_{n,k}\widetilde{R}_n)$. Let us fix such a nontrivial minor of $S(G_{n,k}\widetilde{R}_n)$, by choosing n rows. Observe that this minor is a polynomial in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$, which satisfies the properties of Lemma 10 for $m = n - 1$. Hence there exist

$$((q-1)^2 - 2(n-1)(q-1))^n$$

choices of the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$ in $\mathbb{F}_q^\times$, s.t. the minor evaluated in these choices is nonzero. Hence we get the claim. □

2. case: $n \geq \frac{1}{2}(k^2 + k)$

We state in the following a lower bound on the size of $\mathcal{G}_{G_{n,k}}$.

Theorem 8. Let $\mathbb{F}_q$ be a finite field and $1 \leq k < n \leq q$ be integers, s.t. $n \geq \frac{1}{2}(k^2 + k) = m$ and $q > 2m$. Under the assumption that there exists a nontrivial minor of $S(G_{n,k}\widetilde{R}_n)$, we have the following lower bound on the size of $\mathcal{G}_{G_{n,k}}$.

$$\left| \mathcal{G}_{G_{n,k}} \right| \geq ((q-1)^2 - 2m(q-1))^n.$$

With this we can state the following two corollaries, about the existence of R_n in $\mathcal{G}_{G_{n,k}}$ and the probability that a random $R_n \in A_n$ is in $\mathcal{G}_{G_{n,k}}$.

Corollary 7. We have the existence of R_n in $\mathcal{G}_{G_{n,k}}$. Since $\left| \mathcal{G}_{G_{n,k}} \right| \geq 1$ for $q > 3$.

Corollary 8. The probability of $R_n \in A_n$ to be in $\mathcal{G}_{G_{n,k}}$ is greater than or equal to

$$\frac{((q-1)^2 - 2m(q-1))^n}{(q-1)^{2n}} = \left(1 - \frac{2m}{q-1}\right)^n,$$

where $m = \frac{1}{2}(k^2 + k)$. And we can observe that for fixed n, k this quantity tends to one for $q \to \infty$.

Proof. Define $m = \frac{1}{2}(k^2 + k)$. By assumption there exists a nontrivial minor of $S(G_{n,k}\widetilde{R}_n)$. Observe that this minor is a polynomial in $\mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$, which satisfies the properties of Lemma 10, with total degree $2m(n-1)$ and in each variable of degree at most $2m$. Hence there exist at least

$$((q-1)^2 - 2m(q-1))^n$$

choices of the variables $x_1, \ldots, x_n, y_1, \ldots, y_n$ in $\mathbb{F}_q^\times$, s.t. the minor evaluated in these choices is nonzero. Hence we get the claim. □

Comparing the different cases we can summarize that the probability of $R_n \in A_n$ to be in $\mathcal{G}_{G_{n,k}}$ is greater than or equal to

$$\left(1 - \frac{2m}{q-1}\right)^n,$$

where $m = \min\left\{n, \frac{1}{2}(k^2+k)\right\}$. We can observe that for fixed n, k this quantity tends to one for $q \to \infty$.

Thus we have achieved to show in the McEliece version, under the assumption of the existence of a nontrivial minor, the existence of $R_n \in \mathcal{G}_{G_{n,k}}$ for each $1 \leq k < n \leq q$ with $q > 2n-2$ and each $G_{n,k}$ and that for large q, the probability that $R_n \in A_n$ is in $\mathcal{G}_{G_{n,k}}$ tends to one.

To make the proposed cryptosystem in the McEliece version secure against the distinguisher attack, one could test if the Schur matrix of the public matrix has full rank and then use the matrix Q to scramble the secure code. The complexity of this can be found in the Chapter 7.

Please observe that one could have also used Theorem 6, the Lemma of Schwartz-Zippel.

If we apply this lemma to the case $n = \frac{1}{2}(k^2+k)$, we set $p = \det(S(G_{n,k}\widetilde{R}_n)) \in \mathbb{F}_q[x_1, \ldots, x_n, y_1, \ldots, y_n]$ of total degree $d = 2n(n-1)$ and $S = \mathbb{F}_q^\times$. Then this gives the following result.

$$\mathbb{P}[p(a_1, \ldots, a_n, b_1, \ldots, b_n) = 0] \leq \frac{2n(n-1)}{q-1}.$$

Thus this argument also shows, that for fixed n the probability of $R_n \in A_n$ to be in $\mathcal{G}_{G_{n,k}}$ tends to one for $q \to \infty$.

One has existence of R_n in $\mathcal{G}_{G_{n,k}}$, whenever

$$\frac{2n(n-1)}{q-1} < 1 \Rightarrow 2n(n-1) < q-1.$$

Whereas with our argument we have existence of R_n in $\mathcal{G}_{G_{n,k}}$ for $2n-2 < q$.

Comparing the probability of $R_n \in A_n$ to be in $\mathcal{G}_{G_{n,k}}$ achieved with our argument, and the probability achieved by using the lemma of Schwartz-Zippel, we note that we have a stronger argument. Since for $n \geq 2$

$$\left(1 - \frac{2(n-1)}{q-1}\right)^n > 1 - \frac{2n(n-1)}{q-1}.$$

The reason for this, is that the lemma of Schwartz-Zippel applies to any polynomial and does not use the fact, that p has in each variable degree at most $2(n-1)$.

5.2.4 Experimental Results

In this section we provide some experimental results to the probability bounds, we will refer to the Niederreiter version.

We fix values for q, n, r and a generator matrix $H_{n,r}$ and make a Monte Carlo test, how many $R_n \in A_n$ are in $\mathcal{G}_{H_{n,r}}$. And we compare these results with the provided probability bounds.

In [9] they suggest to use the proposed cryptosystem in the parameters $q = 2^9$ and $q = 2^8$. We also tested the parameters $q = 151$ and $q = 128$.

q	n	r	Monte Carlo test with 1000 tries	probability bound
512	500	250	1	$\geq 1/8$
256	255	100	1	$\geq 1/8$
151	100	50	1	$\geq 1/4$
128	100	50	1	$\geq 1/5$

This result shows, that each invertible row and column weight two matrix we have chosen randomly, satisfies the property, that the public matrix has maximal square code dimension. This especially also shows, for these parameters and for the chosen generator matrices of GRS codes, that the assumption of the nontrivial minor is satisfied.

These experimental results were made with Sage [34]. The used Sage functions can be found in the appendix.

5.3 Extended Distinguisher Attack

The extended distinguisher attack [14] uses that the public code of the Niederreiter version of the BBCRS scheme has unusual small square code dimension, when shortened in a subset of J_1, i.e. positions which come from rows of weight 1, as observed in Proposition 11.

In the proposed cryptosystem we have that $J_1 = \emptyset$ and $J_2 = \{1, \ldots, n\}$. Therefore if the public code is shortened in a subset of J_1, we still have the public code, for which we have shown that with high probability the square code has maximal dimension. Therefore an attacker can not distinguish with this method if the secret code was chosen random or a GRS code.

In the extended distinguisher attack, if the resulting code C, which is in the case of the proposed cryptosystem the public code, contains remaining positions of weight 2, the code gets punctured in the positions of J_2. Which in the case of the proposed cryptosystem would mean to delete all coordinates of the codewords, and this clearly provides no information to an attacker.

5.4 ISD Attack

The ISD attack is a non-structural attack, which aims to decode a random code without exploiting any structural property of the code, hence it is non-polynomial in the dimension of the code. We refer to the ISD attack presented in [31], which is a generalization of Stern's algorithm [38].

To secure the cryptosystem against the ISD attack, one has to choose the parameters large enough, such that a given work factor for the ISD attack is achieved, which is considered to be unfeasible and hence makes the system secure. This clearly will influence the key size of the cryptosystem. In Chapter 7 we will use [32] to give an estimate on the cost of the ISD attack for the parameters we suggested in Section 5.2.4 and we estimate the key size for a given work factor for the ISD attack.

6 Vulnerabilities

As for the security against the distinguisher attack [18, 13] we were only able to prove that the probability of the scrambling matrices to avoid this attack is high, there is also a small probability that the attack could still reveal information of the secret code. This result was also stated under an assumption, which is not proven yet. In Chapter 7 we provide the complexity cost of testing that the Schur matrix of the public matrix is of full rank. With this one can test the security against the distinguisher attack, but this certainty comes with a cost of complexity.

In the Niederreiter version, we have that the public matrix is a sum of two GRS codes, this could exploit some information on the secret GRS code, if one would be able to extend the attack of Sidelnikov and Shestakov [37].

7 Complexity and Key Size

We will estimate in this chapter the key size and the complexity of the cryptosystem proposed in [9], which we presented in Chapter 4.

As stated in the BBCRS Scheme [4], increasing the choice of m increases the key size and increasing z increases the complexity of the decryption. For the following computations and comparisons of key sizes and complexity we will refer to the Niederreiter version.

7.1 Key Size

The public key of the proposed system is a $r \times n$ matrix over $\mathbb{F}_q$, and the weight of the error vector, which is negligible. We can write the public matrix in systematic form to bring the size down to a $r \times (n-r)$ matrix. Thus we consider the key size to be q^{rn}, resp. q^{rk}.

The key size can be considered as a large when compared to other cryptosystem, such as RSA with a key size of $2n$, twice the block size.

But the key size of the proposed cryptosystem is considered to be smaller than the key size of the original McEliece system using Goppa codes, for the following reason. To have similar security against a brute-force attack, we need in the Goppa-based system a bigger public matrix, than in the GRS-based system. The intuitional idea of a brute-force attack would be to go through all words which have at most distance t of the received word in $\mathbb{F}_q^n$. Hence there are

$$\sum_{i=0}^{t} \binom{n}{i} (q-1)^i$$

many words to check in a brute-force attack. Where $\binom{n}{i}$ is for the error position and $(q-1)^i$ is for the magnitude of the error. In the Goppa-based system, we usually choose $n = 2^m$, therefore the received codeword is in $\mathbb{F}_2^n$ and there is no influence of the error magnitude. This allows us to take in the proposed cryptosystem smaller key sizes and to get similar security against this brute-force attack as in the original McEliece cryptosystem with larger key size.

Example 3. For example, take in the Goppa-based system

$$n = 2^{10} = 1024,\ k = 524,\ t = 50,$$

and in the proposed cryptosystem

$$q' = 211,\ n' = 210,\ k' = 105,\ t = 52,\ t_{\text{pub}} = 26.$$

Then we get better brute-force security using the proposed cryptosystem, since by the computations above, we have

$$\sum_{i=0}^{26} \binom{210}{i} 210^i > \sum_{i=0}^{50} \binom{1024}{i}.$$

But in the proposed cryptosystem we get smaller key sizes. Since the key size of the Goppa-based system is 2^{k^2}, whereas the key size of the proposed cryptosystem is $q'^{k'^2}$. So in the example we get in the Goppa-based system a 274576 bit key, and in the GRS-based system a 85126 bit key.

The key size of a code-based cryptosystem is highly influenced by the ISD attack, which aims to decode a random code. We use the provided PARI/GP script of [32] to give an estimate on the cost of the ISD attack for the parameters we suggested in Section 5.2.4 and we estimate the key size to achieve a work factor of 2^{80} and 2^{128} for the ISD attack.

q	n	r	Key Size	Cost of ISD in binary logarithm
1024	1000	500	2500000	157
797	790	400	1560000	128
512	500	250	562500	87
457	450	225	455625	80

We want to compare the key size of the proposed cryptosystem with the original McEliece cryptosystem and the BBCRS scheme regarding security against the ISD attack.

In [8] they suggest to use in the original McEliece version the parameters $n = 1632, k = 1269$ in order to reach a work factor greater than 2^{80}, which results in a public key of size 460647 bits. And for a work factor greater than 2^{128}, they suggest the parameters $n = 2960, k = 2288$, which gives a public key of size 1537536 bits.

In [4] they suggest to use the BBCRS scheme with the parameters $q = 347, n = 346, k = 252$ to reach a work factor of 2^{90} and obtain a public key of size 199899 bits. And for a work factor greater than 2^{137}, they suggest to use the parameters $q = 547, n = 546, k = 396$, which results in a public key of size 540267 bits.

Regarding the cost of the ISD attack computed by [32], for the proposed cryptosystem to reach a work factor greater than 2^{80}, one can use for example the parameters $q = 457, n = 450, k = 225$, which gives in the Niederreiter version a public key of size 455625 bits. For a work factor greater than 2^{128}, one can use the parameters $q = 773, n = 770, k = 410$, which results in a public key of size 1476000 bits.

To compare these stated suggestions we provide the following two tables.

For a work factor of 2^{80}

	q	n	k	Key Size
Original McEliece	2048	1632	1269	460647
BBCRS scheme	347	346	252	199899
Proposed Cryptosystem	457	450	225	455625

For a work factor of 2^{128}

	q	n	k	Key Size
Original McEliece	4096	2960	2288	1537536
BBCRS scheme	547	546	396	540267
Proposed Cryptosystem	773	770	410	1476000

We observe that the key sizes of the proposed cryptosystem are quite close to, but smaller than the key sizes of the original McEliece cryptosystem.

7.2 Complexity

We follow the computations of [11], as it is also done in [4]. We will use the following Notation.

In the sequel denote by $\mathcal{M}$ the number of bit operations required to multiply two field elements of $\mathbb{F}_q$ and by $\mathcal{S}$ the number of bit operations required to add two field elements of $\mathbb{F}_q$.

Where the cost $\mathcal{S}$ of one addition over a finite field $\mathbb{F}_q$ is considered equal to $l = \lceil \log_2(q) \rceil$ binary operations and the cost of one multiplication $\mathcal{M}$ equals to $2l$ additions, thus $\mathcal{M} = 2l^2$ binary operations. Inversion over a finite field comes with the same cost as multiplication. If we multiply matrices, we multiply the matrix to each column vector. Whereas right multiplication of a $m \times n$ matrix with a vector with w nonzero elements means to sum w columns, which costs $(w-1)m\mathcal{S}$ binary operations. Similarly left multiplication costs $(w-1)n\mathcal{S}$ binary operations. This quantity must be added with the operations needed to multiply each element of the vector by the corresponding matrix column, further $wm\mathcal{M}$, resp. $wn\mathcal{M}$ binary operations.

We will give in the following an upper bound for total number of bit operations required for the Niederreiter version of the proposed cryptosystem.

Lemma 11. An upper bound for the total number of bit operations required for encryption in the Niederreiter version is given by

$$(t_{\text{pub}} - 1)r\mathcal{S} + t_{\text{pub}} r\mathcal{M}.$$

Proof. For encryption in the Niederreiter version, we need to compute $H'x^T$. Where H' is a $r \times n$ matrix and x^T has weight $t_{\text{pub}} \leq \lfloor \frac{r}{4} \rfloor$. Therefore the encryption step requires $(t_{\text{pub}} - 1)r\mathcal{S} + t_{\text{pub}} r\mathcal{M}$ binary operations. □

Lemma 12. An upper bound for the total number of bit operations required for decryption in the Niederreiter version is given by

$$\mathcal{M}\left\{r^2 + 10t^2 + t(n+9) - n + tn\right\} + \mathcal{S}\left\{r(r-1) + 6t^2 + t(n+1) + (t-1)n\right\}.$$

Proof. For decryption we first need to compute for the given cipher y the quantity Sy, which requires $(r-1)r\mathcal{S} + r^2\mathcal{M}$ binary operations. For syndrome decoding the codeword we refer to the standard GRS syndrome decoding algorithm, whose complexity can be found in [11], which results in

$$\mathcal{S}\left\{6t^2 + t(n+1)\right\} + \mathcal{M}\left\{10t^2 + t(n+9) - n\right\}$$

binary operations. As last step when we are given $Q^T x^T$, which is of weight $t \leq \lfloor \frac{r}{2} \rfloor$, we need to multiply $(Q^T)^{-1}$ by this vector, thus we have further $(t-1)n\mathcal{S} + tn\mathcal{M}$ binary operations. Which gives us a complexity estimation of

$$\mathcal{M}\left\{r^2 + 10t^2 + t(n+9) - n + tn\right\} + \mathcal{S}\left\{r(r-1) + 6t^2 + t(n+1) + (t-1)n\right\}$$

binary operations. □

As comparison we can look at the decryption complexity of the BBCRS scheme [4], which results in

$$\mathcal{M}\left\{(4t(2t+2)+r)\frac{q^z}{2} + 2t^2 + (2n+1)t + r^2 - n\right\} + \mathcal{S}\left\{(2t(2t+1)+r)\frac{q^z}{2} + 2t^2 + (2n-1)t + (r-1)r - n\right\}$$

binary operations. We can observe that this complexity is greater than the decryption complexity of the proposed cryptosystem. Since the BBCRS scheme is proposed for $z \geq 1$, hence $\frac{q^z}{2} \geq 1$. Therefore is the decryption complexity of the new variant an improvement to the BBCRS scheme.

By Lemma 11 and Lemma 12, the complexity of the Niederreiter version of the proposed cryptosystem can be estimated as follows.

Corollary 9. An upper bound for the total number of bit operations required for the Niederreiter version is given by

$$\mathcal{M}\left\{t_{\text{pub}}r + r^2 + 10t^2 + t(n+9) - n + tn\right\} + \mathcal{S}\left\{(t_{\text{pub}} - 1)r + r(r-1) + 6t^2 + t(n+1) + (t-1)n\right\}.$$

7.2.1 Complexity of Testing Security

For the security of the cryptosystem proposed in [9] to be established, we would like to add here the cost of testing if the Schur matrix of the public matrix has full rank.

Lemma 13. An upper bound for the total number of bit operations required for testing the security of the cryptosystem is given by

$$\mathcal{M}\left(nr^2 + 2nr + \sum_{i=0}^{m-2}((n-i)(m-i)) + (n-m+1)\right) + \mathcal{S}\left(n(r-1)r + nr + \sum_{i=0}^{m-2}(n-i)(m-i-1)\right).$$

Proof. First we need to compute the public matrix, which is given by SHQ^T. Where S is a $r \times r$ matrix, H is an $r \times n$ matrix and Q^T is a $n \times n$ matrix, s.t. each column has two nonzero elements. The computation of SH costs $n((r-1)r\mathcal{S} + r^2\mathcal{M})$ binary operations. The computations of $(SH)Q^T$ costs $n(r\mathcal{S} + 2r\mathcal{M})$ binary operations.

As a second step we want to compute the Schur matrix of the $r \times n$ public matrix. For this we need to compute the Schur product of each row with each other row. The cost of computing the Schur product of two vectors of length n is $n\mathcal{M}$. We do this $\frac{1}{2}(r^2 + r)$ many times, resulting in $\frac{1}{2}(r^2 + r)n\mathcal{M}$ binary operations.

As last step we want to use Gauss elimination, to bring the Schur matrix of size $\frac{1}{2}(r^2+r)\times n$ in echelon form. We can assume that $\frac{1}{2}(r^2+r) = m < n$, if this is not the case, we look at the transposed of the Schur matrix. The computation consists of two steps, we consider to be in the $i + 1$th row. The first step is to set the pivot element to one. For this we multiply a scalar to a row, consisting of $(n - i)$ nonzero elements, which results in $(n - i)$ multiplications. As second step we want to set all elements of the column below the pivot to zero, which are $(m - i - 1)$ many elements, this consists in multiplying a row with $(n - i)$ nonzero elements by a scalar and then adding two rows with $(n - i)$ elements, hence $(m - i - 1)$ times $(n - i)$ multiplications and $(n - i)$ additions. We do these steps up to the last row, since in the last row it is enough to do the first step which costs $(n - m + 1)$ multiplications. Hence we get for the Gaussian elimination the following cost.

$$\mathcal{M}\left(\sum_{i=0}^{m-2}((n-i)+(n-i)(m-i-1))+(n-m+1)\right)+ \\ \mathcal{S}\left(\sum_{i=0}^{m-2}(n-i)(m-i-1)\right).$$

Therefore the total complexity for the security test can be estimated as

$$\mathcal{M}\left(nr^2+2nr+\sum_{i=0}^{m-2}((n-i)(m-i))+(n-m+1)\right)+ \\ \mathcal{S}\left(n(r-1)r+nr+\sum_{i=0}^{m-2}(n-i)(m-i-1)\right).$$

□

8 Conclusion

In this Master thesis we presented a new code-based cryptosystem, proposed by Bolkema, Gluesing-Luerssen, Kelley, Lauter, Malmskog and Rosenthal in [9], we analyzed the security of the proposed cryptosystem against the distinguisher attacks based on the Schur product [14, 18, 13] and compared its key size to the original McEliece system.

The key sizes of the proposed cryptosystem are not as low as for example the highly used RSA. But the key sizes are comparable and slightly smaller than in the original McEliece cryptosystem [26] using Goppa codes.

There is still the open question, if one can find a construction of the invertible row and column weight two matrix, which for a given GRS code will satisfy that the public code has maximal dimension under the square code. We remark here that the entries of the row and column weight two matrix must depend on q and the α, β of the secret $\mathrm{GRS}_{n,r}(\alpha, \beta)$ code.

Seeing the proposed cryptosystem as a special case of the BBCRS scheme in [4], an improvement of the proposed cryptosystem could be setting $z = 1$.

We also want to prove the assumption we used in the argument for the security, that there exists a nontrivial minor of the Schur matrix of the public matrix.

But the proposed cryptosystem has another property in his favor, it resists with high probability a distinguisher attack, which is based on the Schur matrix.

There are also other code-based cryptosystems, which have been attacked by similar distinguisher attacks. Janwa and Moreno proposed in [19] the use of AG codes, or codes derived from them by subfield restriction or concatenation, for a code-based cryptosystem.

Couvreur, Márquez-Corbella and Pellikaan presented in [12] an attack on this cryptosystem, by deriving a t-error correcting pair with the aid of a filtration that is based on the Schur product.

One could investigate if one can find a set of scrambling transformations where the best known distinguisher attacks (as described above coming from the Schur product) will fail. This then hopefully will provide post-quantum cryptosystems with reasonable key sizes.

9 Acknowledgments

A special thank goes to Joachim Rosenthal for this opportunity of investigating one of his ideas in a very ongoing topic, for letting enough freedom of how to prove and develop ideas but at the same time being always available for questions.
I would also like to thank Joachim Rosenthal personally for encouraging me and believing in me not only in this master thesis but also for my further career.

I would like to thank Reto Schnyder the co-supervisor of this thesis for correcting many attempts, pointing out missing things and for being always available, it has been of a great help to discuss with you.

I would like to thank Alessandro Neri for helpful discussions and Markus Neumann for his help and effort in the experimental results.

I would like to thank my family and friends for their unconditioned support. They were of great help in times of struggles and I would not have been able to do this thesis without them.

I also want to thank my fellow students. We always had a great atmosphere in the mathematics department and it was always a pleasure to discuss mathematics with you. Without such great company I would not have completed these studies.

And last but not least I want to thank my partner, Giacomo Micheli for his great support and love.

10 Appendix

10.1 Sage Functions

We will provide here the Sage functions used in Section 5.3.4.

```
def check_weight(A,n,ww):
    w=zero_vector(n)
    for i in range(n):
        for j in range(n):
            if(A[i][j]!=0):
                w[i]=w[i]+1
    counter=0
    for l in range(n):
        if(w[l]==ww):
            counter=counter+1
    if(counter==n):
        return True
    if(counter!=n):
        return False


def distinctvector(V):
    x=V.zero_vector()
    F=V.base_field()
    for i in range(1,V.dimension()):
        t=F.random_element()
        while(t in x):
            t=F.random_element()
        x[i]=t
    return x


def construct_G(q,k,n):
    F=GF(q, 'a')
    a=F.multiplicative_generator()
    y=rand_vec(q,n)
    V=VectorSpace(F,n)
    x=distinctvector(V)
    G=zero_matrix(F,k,n)
    for m in range(k):
        G[m]=(y.pairwise_product(vector(F, [x[j]^m for j in range(n)])))
    return G
```

```
def construct_prerequisites(q,n,r, base_path='.'):
    F=GF(q, 'a')
    a=F.multiplicative_generator()
    badn=0
    goodn=0
    H=construct_G(q,r,n)
    S=SymmetricGroup(n)
    f=S.random_element()
    h=S.random_element()
    Q=h.matrix()
    P=f.matrix()
    while(check_weight(P+Q, n,2)== False):
        h=S.random_element()
        Q=h.matrix()
    Q.save(base_path + '/Q.{q}-{n}-{r}-sage-save'.format(q=q,n=n,r=r))
    P.save(base_path + '/P.{q}-{n}-{r}-sage-save'.format(q=q,n=n,r=r))
    H.save(base_path + '/H.{q}-{n}-{r}-sage-save'.format(q=q,n=n,r=r))


import argparse


def rand_vec(q,n):
    x=zero_vector(n)
    F=GF(q, 'a')
    a=F.multiplicative_generator()
    v = vector([randint(0,q-2) for i in range(n)])
    x = vector([a^v[i] for i in range(n)])
    return x


def schur_product(x,y,q):
    F=GF(q, 'a')
    n=x.length()
    z=vector(F, [x[i]*y[i] for i in range(n)])
    return z


def square_matrix(G,q):
    F=GF(q, 'a')
    k=G.nrows()
    n=G.ncols()
    M=zero_matrix(F,0,n)
    for i in range(k):
        Mi=zero_matrix(F,(k-i),n)
        for l in range(k-i):
            Mi[l]=schur_product(G[i],G[l+i],q)
        M=M.stack(Mi)
    return M
```

```
def MonteCarloNiederreiterR(q,n,r, bound, base_dir='.'):
    badn=0
    goodn=0
    Q = load(base_dir + '/Q.{q}-{n}-{r}-sage-save'.format(q=q,n=n,r=r))
    P = load(base_dir + '/P.{q}-{n}-{r}-sage-save'.format(q=q,n=n,r=r))
    H = load(base_dir + '/H.{q}-{n}-{r}-sage-save'.format(q=q,n=n,r=r))
    print "Loading done"
    m=min(n,1/2*(r^2+r))
    for l in range(bound):
        v=rand_vec(q,n)
        w=rand_vec(q,n)
        A=diagonal_matrix(v);
        B=diagonal_matrix(w);
        R=P*A+Q*B
        if(R.is_invertible()):
            GG=H*R
            SM=square_matrix(GG,q)
            r=SM.rank()
            if(r==m):
                goodn+=1
            else:
                badn+=1
    return badn, goodn


if __name__=='__main__':
   parser = argparse.ArgumentParser()
   parser.add_argument('q', help='base field dim', type=int)
   parser.add_argument('n', help='number of columns', type=int)
   parser.add_argument('r', help='number of rows', type=int)
   parser.add_argument('bound', help='number of monte carlo trials', type=int)
   parser.add_argument('run_id', help='identify this run', type=int)
   args = parser.parse_args()
   bad, good = MonteCarloNiederreiterR(args.q, args.n, args.r, args.bound)
   with open('bad-{id}'.format(id=args.run_id), 'w') as f:
           f.write(str(bad)+'\n')
   with open('good-{id}'.format(id=args.run_id), 'w') as f:
           f.write(str(good)+'\n')
```

References

[1] National Security Agency, Cryptography today, August 2015, archived on 23 November 2015, tinyurl.com/SuiteB.

[2] Report on Post-Quantum Cryptography. Technical report, National Institute of Standards and Technology, February 2016. NISTIR 8105.

[3] Use of Public Standards for the Secure sharing of Information Among National Security Systems. Technical report, Committee on National Security Systems, July 2015. CNSS Advisory Memorandum.

[4] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, D. Schipani. Enhanced Public Key Security for the McEliece Cryptosystem. *Journal of Cryptology, volume 29*. Pages: 1-27, 2016.

[5] T. P. Berger, P. Loidreau. How to Mask the Structure of Codes for a Cryptographic Use. *Designs, Codes and Cryptography, volume 35*. Pages: 63–79, 2005.

[6] E. Berlekamp, R. McEliece, H. van Tilborg. On the Inherent Intractability of certain Coding Problems. *IEEE Transactions on Information Theory, volume 24, issue 3*. Pages: 384–386, 1978.

[7] D. J. Bernstein, J. Buchmann, E. Dahmen. *Post-Quantum Cryptography*. Springer, 2009.

[8] D. J. Bernstein, T. Lange, C. Peters. Attacking and Defending the McEliece Cryptosystem. *Post-Quantum Cryptography, Lecture Notes in Computer Science, volume 5299*. Pages: 31–46, 2008.

[9] J. Bolkema, H. Gluesing-Luerssen, C. A. Kelley, K. Lauter, B. Malmskog, J. Rosenthal. Variations of the McEliece Cryptosystem. arXiv:1612.05085, 2016.

[10] I. Cascudo, R. Cramer, D. Mirandola, G. Zémor. Squares of Random Linear Codes. *IEEE Transactions on Information Theory, volume 61, issue 3*. Pages: 1159 - 1173, 2015.

[11] N. Chen, Z. Yan. Complexity Analysis of Reed-Solomon Decoding over GF (2^m) without using Syndromes. *EURASIP Journal on Wireless Communications and Networking, Article ID 843634*. 2008.

[12] A. Couvreur, I. Márquez-Corbella, R. Pellikaan. Cryptanalysis of Public-Key Cryptosystems that use Subcodes of Algebraic Geometry Codes. *Coding Theory and Applications, CIM Series in Mathematical Sciences 3*. Pages 133–140, Springer, 2015.

[13] A. Couvreur, P. Gaborit, V. Gauthier-Umaña, A. Otmani, J.-P. Tillich. Distinguisher-based Attacks on Public-Key Cryptosystems using Reed-Solomon Codes.*Designs, Codes and Cryptography, volume 73*. Pages: 641-666, 2014.

[14] A. Couvreur, V. Gauthier-Umaña, A. Otmani, J.-P. Tillich. A Polynomial-Time Attack on the BBCRS Scheme. *Public-key cryptography—PKC 2015, volume 9020 of Lecture Notes in Computer Science*. Pages 175–193, Springer, 2015.

[15] J. Ding, J. E. Gower, D. S. Schmidt. *Multivariate Public Key Cryptosystems, volume 25 of Advances in Information Security*. Springer, 2006.

[16] M. Elia, J. Rosenthal, D. Schipani. On the Decoding Complexity of Cyclic Codes Up to the BCH Bound. *2011 IEEE International Symposium on Information Theory Proceedings*. Pages: 835 - 839, 2011.

[17] J.-C. Faugère, V. Gauthier, A. Otmani, L. Perret, J.-P. Tillich. A Distinguisher for High Rate McEliece Cryptosystems. *In Proceedings of the Information Theory Workshop 2011*. Pages: 282–286, 2011.

[18] V. Gauthier-Umaña, A. Otmani, J.-P. Tillich. A Distinguisher-based Attack on a Variant of McEliece's Cryptosystem Based on Reed-Solomon Codes. http://arxiv.org/abs/1204.6459, 2012.

[19] H. Janwa and O. Moreno. McEliece Public Cryptosystem using Algebraic-Geometric Codes. *Designs, Codes and Cryptography, volume 8* Pages: 293–307, 1996.

[20] K. Lee. Interpolation-based Decoding of Alternant Codes. arXiv:cs/0702118, 2007.

[21] Y. X. Li, R. H. Deng, X. Wang. On the Equivalence of McEliece's and Niederreiter's Public-Key Cryptosystems. *IEEE Transactions on Information Theory, volume 40*. Pages: 271-273, 1994.

[22] F. J. MacWilliams, N. J. A. Sloane. The Theory of Error-Correcting Codes, volume 16. 1978.

[23] I. Márquez-Corbella, E. Martínez-Moro, R. Pellikaan. The non-gap Sequence of a Subcode of a Generalized Reed–Solomon Code. *Designs, Codes and Cryptography, volume 66* Pages: 317-333, 2013.

[24] I. Márquez-Corbella, E. Martínez-Moro, R. Pellikaan. Evaluation of Public-key Cryptosystems Based on Algebraic Geometry Codes. *Proceedings of the Third International Castle Meeting on Coding Theory and Applications*. Pages 199–204, 2011.

[25] I. Márquez-Corbella, R. Pellikaan. Error-Correcting Pairs for a Public-key Cryptosystem. *Proceedings Code-based Cryptography Workshop (CBC 2012)*. 2012.

[26] R. McEliece. A Public-Key Cryptosystem based on Algebraic Coding Theory. *DSN Progress Report, volume 42*. Pages: 114-116, 1978.

[27] K. S. Miller. On the Inverse of the Sum of Matrices. *Mathematics Magazine, volume 54*. Pages: 67-72, 1981.

[28] H. Niederreiter. Knapsack-type Cryptosystems and Algebraic Coding Theory. *Problems of Control and Information Theory, volume 15*. Pages: 159-166, 1986.

[29] C. Peikert. A Decade of Lattice Cryptography. Cryptology ePrint Archive, Report 2015/939, 2015. `http://eprint.iacr.org/2015/939`.

[30] R. Pellikaan. On Decoding by Error Location and Dependent Sets of Error Positions. *Discrete Mathematics, volumes 106–107*. Pages: 368–381, 1992.

[31] C. Peters. Information-Set Decoding for Linear Codes over $\mathbb{F}_q$. *Post-Quantum Cryptography, Lecture Notes in Computer Science, volume 6061*. Pages: 81-94, Springer, 2010.

[32] C. Peters. 2010, `http://christianepeters.wordpress.com/publications/tools/`.

[33] R. L. Rivest, A. Shamir, L. Adleman. Patent US4405829: Cryptographic Communications System and Method. Massachusetts Institute of Technology, 1983.

[34] *SageMath, the Sage Mathematics Software System (Version 7.1)*, The Sage Developers, 2016, `http://www.sagemath.org`.

[35] J.T. Schwartz. Fast Probabilistic Algorithms for Verification of Polynomial Identities. *Journal of the ACM, volume 27, issue 4*. Page: 701–717, 1980.

[36] P. Shor. Algorithms for Quantum Computations: Discrete Logarithms and Factoring. *Proceedings of the 35th Annual Symposium on the Foundations of Computer Science*. Pages: 124-134, 1994.

[37] V. Sidelnikov, S. Shestakov. On Insecurity of Cryptosystems based on Generalized Reed-Solomon Codes. *Discrete Math Appl., volume 2*. Pages: 439-444, 1992.

[38] J. Stern. A Method for Finding Codewords of small Weight. *Coding Theory and Applications, LNCS, volume 388*. Pages: 106-113, Springer, 1989.

[39] J.H. van Lint. Introduction to Coding Theory, second edition. Springer, 1992.

[40] C. Wieschebrink. Cryptanalysis of the Niederreiter Public Key Scheme Based on GRS Subcodes. *Post-Quantum Cryptography, Third International Workshop, PQCrypto 2010, volume 6061 of Lecture Notes in Computer Science*. Pages: 61–72, 2010.