Security of cryptographic protocols based on coding theory

Herve Tale Kalachi

Normandie Université

THÈSE EN CO - TUTELLE INTERNATIONALE

Pour obtenir le diplôme de doctorat

Spécialités : INFORMATIQUE & MATHÉMATIQUES

Préparée au sein de l'Université de Rouen Normandie

et de l'Université de Yaoundé 1, Cameroun

Sécurité de Protocoles Cryptographiques Fondés sur la Théorie des Codes Correcteurs d'Erreurs

Présentée et soutenue par
Hervé TALE KALACHI

Thèse soutenue publiquement le 05 juillet 2017 devant le jury composé de		
M. Thierry BERGER	Professeur Emérite, Université de Limoges, XLIM	Rapporteur
M. Alain COUVREUR	Chargé de Recherche, INRIA Saclay, Laboratoire d'Informatique de l'Ecole Polytechnique (LIX)	Examinateur
M. Pierre LOIDREAU	Assimilé Maitre de Conférence HDR, Université de Rennes 1 & DGA - Rennes	Rapporteur
M. Selestin NDJEYA	Université de Yaoundé 1	Examinateur
M. Ayoub OTMANI	Professeur, Université de Rouen Normandie, LITIS	Examinateur
M. Jean-Pierre TILLICH	Directeur de Recherche, INRIA Paris	Examinateur

Thèse dirigée par

M. Ayoub OTMANI — Université de Rouen Normandie
M. Selestin NDJEYA & M. Marcel TONGA — Université de Yaoundé 1

UNIVERSITÉ DE ROUEN NORMANDIE

MIIΣ

litis

CRFD-STG

URFD-MIBA

Security of cryptographic protocols based on coding theory

Hervé Talé Kalachi

À mon père et ma mère.

Remerciements

J'ai passé de magnifiques années de thèse grâce à une combinaison d'efforts, d'assistances et de bonnes atmosphères dont j'ai bénéficié autant en France qu'au Cameroun.
Je souhaite premièrement dire merci à mon directeur de thèse à l'université de Rouen, Monsieur Ayoub OTMANI qui a accepté de codiriger cette thèse à un moment où je m'apprêtais à ouvrir la porte au découragement. Il faut dire que ce n'était que le début du commencement car, j'ai encore en souvenir la période difficile de recherche de financement avec les multiples lettres et attestations qu'il a dû écrire pour m'aider à trouver un financement pour ma mobilité. Je suis reconnaissant pour les multiples discussions enrichissantes que nous avons eu pendant mes différents séjours en France. Sa maîtrise de la cryptographie basée sur les codes, son sens aiguisé de l'orientation et sa précision dans la nature et la qualité des questions posées sont des atouts qui nous ont menés droit vers les résultats obtenus dans cette thèse.
Merci à tous les membres de l'équipe "Combinatoire et Algorithmique" du LITIS pour la bonne atmosphère dont j'ai bénéficié dans les couloirs, pendant les pauses café ainsi que les déjeuners en commun. Un merci particulier à mon ami Vlad DRAGOI pour les multiples discussions que nous avons eu ensemble et qui ont été un plus dans ma culture en cryptographie basée sur les codes.
Merci à tous les doctorants de l'équipe pour la bonne ambiance que nous avons partagé durant ces années de thèse dans des parties déstressantes de "Bataille Corse", de "King of Tokyo" et de "Tantrix".

Au Cameroun, je souhaite premièrement remercier mon directeur de thèse Monsieur Sélestin NDJEYA pour les multiples encouragements et conseils, et surtout pour le solide fondement en théorie des codes algébriques qu'il m'a transmis à travers ces enseignements depuis le Master, et dont j'ai pris goût au point de m'intéresser aux applications en cryptographie, premièrement dans mon mémoire de Master qu'il a dirigé et finalement dans mon projet de thèse dont les fruits sont présents dans ce manuscrit.
J'aimerai aussi remercier mon superviseur Monsieur Marcel TONGA pour son soutien et ses conseils, sans oublier les bases de logique et algèbre universelle qu'il m'a transmis à travers ses enseignements de Master et qui m'ont été d'une grande importance durant cette thèse.
Merci à tous les membres de l'Équipe de Recherche en Algèbre et Logique (ERAL) de l'université de Yaoundé 1 pour les multiples échanges, questions et discussions

pendant et après les séminaires. Un merci particulier à Monsieur Célestin NKUIMI pour sa rigueur. Le seul fait de savoir qu'il sera présent lors d'un exposé pousse à être clair et précis dans la préparation.
Un énorme merci au Pôle de Recherche en Mathématiques et Applications à la Sécurité de l'Information (PRMASI) pour m'avoir soutenu financièrement pour plusieurs séjours en France et pour plusieurs conférences et colloques. J'en profite aussi pour remercier tous les membres de PRMASI en commençant par le responsable Tony EZOME, Emmanuel FOUOTSA et tous les autres ; vous avez tous été une source d'inspiration et de motivation pour moi ces dernières années.
Un spécial merci à mon chers aîné et ami Emmanuel FOUOTSA qui a été pour moi le pont entre le Cameroun et la France en me mettant en contact avec plusieurs chercheurs Français de haut niveau et notamment Monsieur Ayoub OTMANI.
J'adresse également mes sincères remerciements au service de coopération et d'action culturelle de l'ambassade de France au Cameroun pour m'avoir financé deux magnifiques séjours de quatre mois en France.
Je remercie Monsieur Thierry BERGER d'avoir accepté de rapporter ce document. Je suis également très honoré d'avoir eu Monsieur Pierre LOIDREAU comme rapporteur de ma thèse.
Pour finir, un énorme merci à mon épouse Sandrine et à mon fils Nell pour leur amour et leur soutien pendant cette période de mobilité et d'instabilité.

Abstract

Contrary to the cryptosystems based on number theory, the security of cryptosystems based on error correcting codes appears to be resistant to the emergence of quantum computers. Another advantage of these systems is that the encryption and decryption are very fast, about five times faster for encryption, and 10 to 100 times faster for decryption compared to RSA cryptosystem.

Nowadays, the interest of scientific community in code-based cryptography is highly motivated by the latest announcement of the National Institute of Standards and Technology (NIST). They initiated the Post-Quantum cryptography Project which aims to define new standards for quantum resistant cryptography and fixed the deadline for public key cryptographic algorithm submissions for November 2017. This announcement motivates to study the security of existing schemes in order to find out whether they are secure. This thesis thus presents several attacks which dismantle several code-based encryption schemes.

We started by a cryptanalysis of a modified version of the Sidelnikov cryptosystem proposed by Gueye and Mboup [GM13] which is based on Reed-Muller codes. This modified scheme consists in inserting random columns in the secret generating matrix or parity check matrix. The cryptanalysis relies on the computation of the square of the public code. The particular nature of Reed-Muller which are defined by means of multivariate binary polynomials, permits to predict the values of the dimensions of the square codes and then to fully recover in polynomial time the secret positions of the random columns. Our work shows that the insertion of random columns in the Sidelnikov scheme does not bring any security improvement.

The second result is an improved cryptanalysis of several variants of the GPT cryptosystem which is a rank-metric scheme based on Gabidulin codes. We prove that any variant of the GPT cryptosystem which uses a right column scrambler over the extension field as advocated by the works of Gabidulin *et al.* [Gab08, GRH09, RGH11] with the goal to resist Overbeck's structural attack [Ove08], are actually still vulnerable to that attack. We show that by applying the Frobenius operator appropriately on the public key, it is possible to build a Gabidulin code having the same dimension as the original secret Gabidulin code, but with a lower length. In particular, the code obtained by this way corrects less errors than the secret one but its error correction capabilities are beyond the number of errors added by a sender, and consequently an attacker is able to decrypt any ciphertext with this degraded Gabidulin code. We also considered the case where an isometric transformation is applied in conjunction with a right column scrambler which has its entries in the extension field. We proved that this protection is useless both in terms of performance and security. Consequently, our results show that all the existing techniques aiming to hide the inherent algebraic structure of Gabidulin codes have failed.

To finish, we studied the security of the Faure-Loidreau encryption scheme [FL05] which is also a rank-metric scheme based on Gabidulin codes. Inspired by our precedent work and, although the structure of the scheme differs considerably from the classical setting of the GPT cryptosystem, we show that for a range of parameters, this scheme is also vulnerable to a polynomial-time attack that recovers the private key by applying Overbeck's attack on an appropriate public code. As an example we break in a few seconds parameters with 80-bit security claim.

Résumé

Contrairement aux protocoles cryptographiques fondés sur la théorie des nombres, les systèmes de chiffrement basés sur les codes correcteurs d'erreurs semblent résister à l'émergence des ordinateurs quantiques. Un autre avantage de ces systèmes est que le chiffrement et le déchiffrement sont très rapides, environ cinq fois plus rapide pour le chiffrement, et 10 à 100 fois plus rapide pour le déchiffrement par rapport à RSA. De nos jours, l'intérêt de la communauté scientifique pour la cryptographie basée sur les codes est fortement motivé par la dernière annonce de la "National Institute of Standards and Technology" (NIST), qui a récemment initié le projet intitulé "Post-Quantum cryptography Project". Ce projet vise à définir de nouveaux standards pour les cryptosystèmes résistants aux attaques quantiques et la date limite pour la soumission des cryptosystèmes à clé publique est fixée pour novembre 2017. Une telle annonce motive certainement à proposer de nouveaux protocoles cryptographiques basés sur les codes, mais aussi à étudier profondément la sécurité des protocoles existants afin d'écarter toute surprise en matière de sécurité.

Cette thèse suit cet ordre d'idée en étudiant la sécurité de plusieurs protocoles cryptographiques fondés sur la théorie des codes correcteurs d'erreurs.

Nous avons commencé par l'étude de la sécurité d'une version modifiée du cryptosystème de Sidelnikov, proposée par Gueye et Mboup [GM13] et basée sur les codes de Reed-Muller. Cette modification consiste à insérer des colonnes aléatoires dans la matrice génératrice (ou de parité) secrète. La cryptanalyse repose sur le calcul de carrés du code public. La nature particulière des codes de Reed-Muller qui sont définis au moyen de polynômes multivariés binaires, permet de prédire les valeurs des dimensions des codes carrés calculés, puis permet de récupérer complètement en temps polynomial les positions secrètes des colonnes aléatoires. Notre travail montre que l'insertion de colonnes aléatoires dans le schéma de Sidelnikov n'apporte aucune amélioration en matière de sécurité.

Le résultat suivant est une cryptanalyse améliorée de plusieurs variantes du cryptosystème GPT qui est un schéma de chiffrement en métrique rang utilisant les codes de Gabidulin. Nous montrons qu'en utilisant le Frobenius de façon appropriée sur le code public, il est possible d'en extraire un code de Gabidulin ayant la même dimension que le code de Gabidulin secret mais avec une longueur inférieure. Le code obtenu corrige ainsi moins d'erreurs que le code secret, mais sa capacité de correction d'erreurs dépasse le nombre d'erreurs ajoutées par l'expéditeur et par conséquent, un attaquant est capable de déchiffrer tout texte chiffré, à l'aide de ce code de Gabidulin dégradé. Nos résultats montrent qu'en fin de compte, toutes les techniques existantes visant à cacher la structure algébrique des codes de Gabidulin ont échoué.

Enfin, nous avons étudié la sécurité du système de chiffrement de Faure-Loidreau [FL05] qui est également basé sur les codes de Gabidulin. Inspiré par les travaux précédents et, bien que la structure de ce schéma diffère considérablement du cadre

classique du cryptosystème GPT, nous avons pu montrer que ce schéma est également vulnérable à une attaque polynomiale qui récupère la clé privée en appliquant l'attaque d'Overbeck sur un code public approprié. Comme exemple, nous arrivons en quelques secondes à casser les paramètres qui ont été proposés comme ayant un niveau de sécurité de 80 bits.

Contents

Chapter 1

Introduction

Cryptography is the field of research in which the techniques of setting up secure communications (in the presence of adversaries) are studied. Nowadays, it is undoubtedly present everywhere. Financial transactions, e-commerce and military applications are just few examples that demonstrate the enormous importance of cryptography in a modern world.

1.1 Motivation

Given this enormous importance of cryptography, many researchers have devoted a great deal of time and effort to propose and analyze cryptographic systems that can be both efficient and secure. Cryptosystems based on number theory (integer factorization and discrete logarithm) such as RSA [RSA78] and elliptic curves cryptography, have been good candidates during several decades and remain widely deployed in practice since they offer a good compromise between efficiency and security. Nevertheless, the existence of sub-exponential algorithms [BGJT14] and polynomial quantum algorithms [Sho94, Sho97] that solve these number theory problems are important facts that make the systems from number theory less and less attractive.

The situation is different for code based cryptography.

Code based cryptography was introduced since 1978 by McEliece [McE78] who was the first to present a cryptosystem based on error-correcting codes. The public key is formed with a matrix $\boldsymbol{G}_{\text{pub}}$ which is obtained by a product of three matrices $\boldsymbol{S}$, $\boldsymbol{G}'$ and $\boldsymbol{P}$. The security of the scheme build is based on two problems: the difficulty of decoding a random linear code [BMvT78] and the difficulty of recovering a decoding algorithm from a public matrix representation of a binary Goppa code. The second assumption was reformulated in a more formal way by stating that there is no polynomial-time algorithm that distinguishes between a random matrix and a generating matrix of a binary Goppa code [CFS01, Sen02]. The scheme disposes of various advantages:

- The encryption and decryption are very fast, about five times faster for encryption, and 10 to 100 times faster for decryption compared to RSA cryptosystem.

- Contrary to the cryptosystems based on number theory, the security of this cryptosystem appears to be resistant to the emergence of quantum computers [Sho94, Sho97].

Although it is efficient, the McEliece cryptosystem came with a big disadvantage: the size of the public keys is about five hundred thousand bits. Several authors have followed the idea of McEliece by trying to solve the problem of key sizes.

Nowadays, the interest of the scientific community in code-based cryptography is highly motivated by the latest announcement of the National Institute of Standards and Technology (NIST). They initiated the Post-Quantum cryptography Project which aims to define new standards for quantum resistant cryptography and fixed the deadline for public key cryptographic algorithm submissions for November 2017 (NIST-PQcrypto Project). This announcement motivates to study the security of existing schemes in order to find out whether they are secure. This thesis thus presents several attacks which dismantle several code-based encryption schemes.

1.2 Previous Works

In order to solve the problem of enormous key size in the McEliece encryption scheme, several authors proposed to replace the family of Goppa codes with another family of codes. The first to propose such an idea is Harald Niedereiter [Nie86] who proposed in 1986 the use of generalized Reed-Solomon codes. However, this was shown six years later to be insecure by Sidelnikov and Shestakov [SS92]. In 1994, Sidelnikov [Sid94] also proposed the use of Reed-Muller codes, but the results presented in [MS07, CB13] show that this variant is not secure. Several papers also follow this idea by proposing the use of another family of codes. Janwa and Moreno [JM96] suggested the use of Algebraic-geometry codes, but this turned out to be insecure [FM08, CMCP14]. Monico, Rosenthal and Shokrollahi proposed and analyzed a variant using low density parity check codes in [MRAS00]. Bernstein, Lange and Peters [BLP10, BLP11] proposed the use of Wild Goppa codes. Srivastava codes were proposed in [Per12] by Persichetti. In [LJ12], Londahl and Johansson proposed the use of convolutional codes, but an efficient attack by Landais and Tillich [LT13] was proposed on this variant only one year later. Polar codes and subcodes of polar codes were also proposed in [SK14, HSEA14], but the variant with polar codes was completely broken in [BCD+16].

During these last decades, several authors have proposed to consider more structured codes. The common idea is to focus on codes equipped with a non-trivial permutation group.[1] This is the case for example of Gaborit [Gab05] who proposed to use quasi-cyclic BCH codes. His work was followed by Berger, Cayrel, Gaborit and Otmani's paper [BCGO09] which used quasi-cyclic alternant codes and the paper of Misoczki and Barreto [MB09] who proposed quasi-dyadic Goppa codes. The algebraic attack given in [FOPT10] succeeds in breaking most of the parameters of

[1] The permutation group of a code is the set of permutations leaving globally invariant the code.

[BCGO09, MB09]. It makes use of the fact that the underlying codes which are alternant codes come with an algebraic structure. It allows a cryptanalysis consisting in setting up a polynomial system and then solving it with Gröbner bases techniques. In the very specific case of [BCGO09, MB09], the quasi-cyclic and quasi-dyadic structures allow a huge reduction of the number of variables. Recently, the attack was further improved against [MB09] by exploiting more efficiently the underlying Goppa structure [FOP+14, FOP+16b].

Although it does not undermine the security of the McEliece scheme, the apparition of algebraic attacks [FOPT10] shows however the importance of finding a better hiding of the structure of the codes. A possible solution would be to change the description of the scheme by inserting some randomness. Berger-Loidreau's paper [BL05] is probably the first attempt towards this objective. The authors suggested to add random rows to the description of the codes. They applied this to Niederreiter encryption scheme [Nie86] instantiated with generalised Reed-Solomon codes. The goal was to come up with a protection against Sidelnikov and Shestakov [SS92]. But Wieshebrink's paper shows that component-wise product of codes [Wie10] enables to break Berger-Loidreau's scheme.

Another simple example would be to insert random columns in the secret matrix. Several authors [Wie06b, GM13] have indeed proposed this technique to avoid structural attacks on similar versions of the McEliece cryptosystem. This kind of modification was proposed for the first time by Wieschebrink in [Wie06b]. Its primary goal was to avoid the Sidelnikov-Shestakov attack [SS92] on the McEliece cryptosystem using generalized Reed-Solomon codes. Although this proposal had effectively avoided the original attack, recent studies have shown that in that case of generalized Reed-Solomon codes, the random columns can be found through considerations of the dimensions of component-wise product of codes [GOT12b, GOT12a, CGG+14]. This insertion of random columns in the secret matrix was also proposed by Gueye and Mboup [GM13] in the case of Reed-Muller codes, with the aim to prevent the key-recovery attacks of [MS07, CB13].

We emphasize that all the variants mentioned above are in Hamming metric. Another variant proposed for the first time in 1991 consists in using codes with another metric, namely the "rank-metric".

The first rank-metric scheme was proposed in [GPT91] by Gabidulin, Paramonov and Tretjakov and is now called the GPT cryptosystem. This scheme can be seen as an analogue of the McEliece scheme public key cryptosystem based on the class of Gabidulin codes. An important operation in the key generation of the GPT cryptosystem is the "hiding" phase where the secret generator matrix $\boldsymbol{G}$ undergoes a transformation to mask the inherent algebraic structure of the associated Gabidulin code. This transformation is a probabilistic algorithm that adds some randomness to its input. Originally, the authors in [GPT91] proposed to use a *distortion* transformation that takes $\boldsymbol{G}$ and outputs the public matrix $\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{G} + \boldsymbol{X})$ where $\boldsymbol{X}$ is a random matrix with a prescribed rank $t_{\boldsymbol{X}}$ and $\boldsymbol{S}$ is an invertible matrix. The presence of a distortion matrix has however an impact: the sender has to add an error vector whose rank weight is $t_{\text{pub}} = t - t_{\boldsymbol{X}}$ where t is the error correction

capability of the secret underlying Gabidulin code. Hence, roughly speaking, the hiding phase publishes a degraded code in terms of error correction.

Gabidulin codes are often seen as equivalent of Reed-Solomon codes in the Hamming metric and, like them, are highly structured. That is the reason why their use in the GPT cryptosystem has been the subject of several attacks. Gibson was the first to prove the weakness of the system through a series of successful attacks [Gib95, Gib96]. Following these failures, the first works which modified the GPT scheme to avoid Gibson's attack were published in [GO01, GOHA03]. The idea is to hide further the structure of Gabidulin code by considering isometries for the rank metric. Consequently, a *right column scrambler* $\boldsymbol{P}$ is introduced which is an invertible matrix with its entries in the base field $\mathbb{F}_q$ while the ambient space of the Gabidulin code is $\mathbb{F}_{q^m}^n$. But Overbeck designed in [Ove05b, Ove05a, Ove08] a more general attack that dismantled all the existing modified GPT cryptosystems. His approach consists in applying an operator Λ_i which applies i times the Frobenius operation on the public generator matrix $\boldsymbol{G}_{\text{pub}}$. Overbeck observed that the dimension increases by 1 each time the Frobenius is applied. He then proved that by taking $i = n - k - 1$ the codimension becomes 1 if k is the rank of $\boldsymbol{G}_{\text{pub}}$ (which is also the dimension of the associated Gabidulin code). This phenomenon is a clearly distinguishing property of a Gabidulin code which cannot be encountered for instance with a random linear code where the dimension would increase by k for each use of the Frobenius operator.

Overbeck's attack uses crucially two important facts, namely the column scrambler matrix $\boldsymbol{P}$ is defined on the based field $\mathbb{F}_q$ and the codimension of $\Lambda_{n-k-1}(\boldsymbol{G}_{\text{pub}})$ is equal to 1. Several works then proposed to resist to this attack either by taking a special distortion matrix so that the second property is not true as in [Loi10, RGH10], or by taking a column scrambler matrix defined over the extension field $\mathbb{F}_{q^m}$ as in [Gab08, GRH09, RGH11].

Besides the McEliece setting used with Gabidulin codes, Faure and Loidreau proposed in [FL05] another approach for designing rank-metric encryption scheme based on Gabidulin codes. The structure of the scheme differs considerably from the classical McEliece setting (there is no masking phase of the Gabidulin code used) and it was supposed to be secure under the assumption that the problem of the *linearized polynomial reconstruction*[2] is intractable. This scheme follows the works done in [AF03, AFL03] where a public-key encryption scheme is defined that relies on the *polynomial reconstruction* problem which corresponds to the decoding problem of Reed-Solomon codes. The Polynomial Reconstruction (PR) consists of solving the following problem: *given two n-tuples $(z_1, \ldots, z_n)$ and $(y_1, \ldots, y_n)$ and parameters $[n, k, w]$, recover all polynomials P of degree less than k such that $P(z_i) = y_i$ for at most w distinct indices $i \in \{1, \ldots, n\}$*. The public key is then a noisy random codeword from a Reed-Solomon code where the (Hamming) weight of the error is greater than the decoding capability of the Reed-Solomon code. However the schemes of [AF03, AFL03] have undergone polynomial-time attacks in [Cor04, KY04]. The authors in [FL05] proposed an analog of Augot-Finiasz scheme [AF03]

[2] In [FL05] the problem is termed as p-polynomial reconstruction problem.

but in the rank-metric context. The security of [FL05] is related to the difficulty of solving p-polynomial reconstruction corresponding actually to the decoding problem of a Gabidulin code beyond its error-correcting capability. After Overbeck's attack, parameters proposed in [FL05] were updated in [Loi07, Chap. 7] in order to resist it.

1.3 Contribution of this Thesis

This thesis presents several attacks on several code-based encryption schemes.

Our first result shows that, as in the case of Reed-Solomon codes, the component-wise product of codes can be used to distinguish a Reed-Muller code from a random code. As a consequence, we have shown that the modified version of the Sidelnikov cryptosystem proposed by Gueye and Mboup [GM13] is actually insecure. This modified scheme consists of inserting random columns in the secret generating matrix or parity check matrix. The cryptanalysis relies on the computation of the squares of the public code. The particular nature of Reed-Muller codes which are defined by means of multivariate binary polynomials, permits to predict the value of dimension of the square codes and then to fully recover in polynomial time the secret positions of the random columns. Our work shows that the insertion of random columns in the Sidelnikov scheme does not bring any security improvement.

This work was done in collaboration with A. Otmani and was published in the proceedings of the conference C2SI-Berger 2015 [OTK15].

The second result was inspired by the links between generalized Reed-Solomon codes and Gabidulin codes in rank-metric. It appears from the results of Overbeck [Ove08] that the equivalent tool of square-code in rank-metric is the map Λ_i used by Overbeck, but with $i = k - 1$. Overbeck used this map with $i = n - k - 1$ and his attack does not succeed on the recent reparations of the GPT cryptosystem. During our analysis of these recent variants, we were able to prove that any variant of the GPT cryptosystem which uses a right column scrambler over the extension field [Gab08, GRH09, RGH11] as advocated by the works of Gabidulin *et al.* with the goal to resist to Overbeck's structural attack [Ove05b, Ove08] are actually still vulnerable to that attack. We showed that by choosing an appropriate value of i, it is possible to build a Gabidulin code having the same dimension as the original secret Gabidulin code but with a lower length. In particular, the code obtained by this way corrects less errors than the secret one but its error correction capabilities are beyond the number of errors added by a sender, and consequently an attacker is able to decrypt any ciphertext with this degraded Gabidulin code. Our results show that all the existing techniques aiming to hide the inherent algebraic structure of Gabidulin codes have failed. This work was in collaboration with S. Ndjeya and A. Otmani and is now accepted to the Journal *Design, Codes and Cryptography* [OTKN16].

The third step was to study the security of the Faure-Loidreau encryption scheme [FL05] which is also a rank-metric scheme based on Gabidulin codes. Inspired by our precedent work, and even if the structure of the scheme differs considerably from

the classical setting of the GPT cryptosystem, we have shown that for a range of parameters, this scheme is also vulnerable to a polynomial-time attack that recovers the private key by applying Overbeck's attack on an appropriate public code. As an example we break in a few seconds parameters with 80-bit security claim. This result is a joint work with P. Gaborit and A. Otmani and was accepted to the journal *Design, Codes and Cryptography* [GOTK16].

1.4 Structure of this Thesis

The sequel of this thesis contains five chapters organized as follows:

- ⋆ Chapter 2 provides the background for the following chapters of the thesis. In particular, we will gather some tools and notions from cryptography, coding theory and we will close this chapter by a state-of-the-art of code-based cryptography. More precisely, it will be the presentation of the McEliece encryption scheme and some comments on its variants.
- ⋆ Chapter 3 develops a cryptanalysis of the modified version given in [GM13] of the Sidelnikov encryption scheme [Sid94] which is a McEliece-type public key encryption scheme [McE78] based on Reed-Muller codes.
- ⋆ In Chapter 4, for a good understanding of the results of the following chapters, we first present some preliminaries of rank metric codes and rank-based cryptography.
- ⋆ In Chapter 5, we present a new structural attacks on the recent variants of the GPT cryptosystem [Gab08, GRH09, RGH11, RGH10].
- ⋆ Chapter 6 presents the Faure-Loidreau scheme [FL05] and the polynomial-time attack on this scheme. The attack recovers the private key from the public key and is based in part on the security analysis given in [Loi07, Chap. 7]. We show that by applying Overbeck's attack on an appropriate public code an attacker can recover the private key very efficiently, only assuming a mild condition on the code, which was always true in all our experimentations.
- ⋆ Finally, the conclusions and perspectives are given in chapter 7.

Chapter 2

Code Based Cryptography

Introduction

The purpose of this chapter is to present some backgrounds and evolutions of code based cryptography. We start in section 2.1 and section 2.2 with some preliminaries of cryptography and coding theory, before presenting the underlying cryptography and some of its variants in section 2.3.

2.1 Cryptography Background

The concept of cryptography is very old. Basically it refers to the process of converting ordinary message (called plaintext) into unintelligible text (called ciphertext). Despite the evolution of the means of communication, it has always been difficult to guarantee the security of the channel through which a message is transmitted. As soon as one wishes to communicate in a secret way, two problems arise:

- **Confidentiality of the message:** The sender has to ensure himself that only the legitimate receiver will be able to read and understand the message.
- **Integrity of the message:** The legitimate receiver has to ensure himself that the ciphertext has not been subject to a modification by a third-party.

In this thesis, we focus on the first point, namely the confidentiality. It can be guaranteed by an "encryption" process, that will be followed by a "decryption" of the legitimate recipient.

2.1.1 Encryption and Decryption

The encryption is an algorithm that allows to convert a given plaintext into a ciphertext that will be readable only by its legitimate recipient. This conversion is performed by an encryption function parameterized by an *encryption key*. The legitimate receiver can then decrypt the ciphertext by using the decryption function if he knows the corresponding *decryption key*. The set of algorithms that generates

encryption and *decryption keys* both with encryption and decryption algorithms is called a cryptosystem or an encryption scheme.

Definition 2.1 (Cryptosystem)**.** A cryptosystem or encryption scheme can be defined as a tuple $(\mathcal{P}, \mathcal{C}, \mathcal{K}, \mathcal{E}, \mathcal{D})$ that satisfies the following properties:

- $\star$ $\mathcal{P}$ is a set called the "plaintext space".
- $\star$ $\mathcal{C}$ is a set called the "ciphertext space".
- $\star$ $\mathcal{K}_e$ is a set called the "space of encryption keys".
- $\star$ $\mathcal{K}_d$ is a set called the "space of decryption keys".
- $\star$ $\mathcal{E} = \{E_{k_e} \ : \ k_e \in \mathcal{K}_e\}$ is a set of encryption functions $E_{k_e} : \mathcal{P} \to \mathcal{C}$.
- $\star$ $\mathcal{D} = \{D_{k_d} \ : \ k_d \in \mathcal{K}_d\}$ is a set of decryption functions $D_{k_d} : \mathcal{C} \to \mathcal{P}$.

For each $k_e \in \mathcal{K}_e$, there is $k_d \in \mathcal{K}_d$ such that $D_{k_d}(E_{k_e}(\boldsymbol{m})) = \boldsymbol{m}$ for all $\boldsymbol{m}$ in $\mathcal{P}$.

If the decryption key is the same as the encryption key, the scheme is called a *secret key cryptosystem* since the keys must be kept secret to ensure the confidentiality. Else, if the encryption key can be published without jeopardizing the confidentiality of the decryption key, the system is called a *public key cryptosystem*. All the cryptosystems mentioned in this thesis are *public key cryptosystems*.

2.1.2 Public Key Encryption Scheme

The first public key encryption scheme was published in 1976 by Whitfield Diffie and Martin Hellman [DH76]. It represents any cryptosystem that uses pairs of keys: public keys (or encryption keys) which may be disseminated widely, and private keys (or decryption keys) which are known only to the owner. The basic concept of public key cryptosystems is the notion of "trapdoor function".

Definition 2.2 (One-way function)**.** A one-way function is a function that is easy to compute on every input, but hard to invert (given the image of a random input).

Definition 2.3 (Trapdoor function)**.** A trapdoor function is a one-way function with a "trapdoor" t that allows to easily invert.

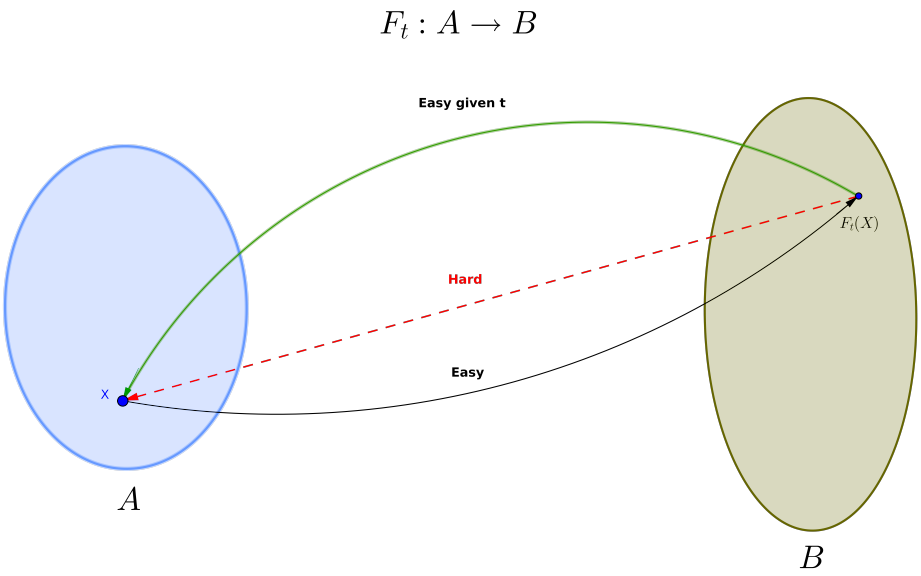

Figure 2.1 – Trapdoor function

Theoretically, it is easy to use a trapdoor function to instantiate a public-key cryptosystem. Let's suppose for example that $F_t : A \longrightarrow B$ is a trapdoor function with trapdoor t. F_t can be used as encryption (or public) key, t as the secret key. One can then encrypt a message $X \in A$ by computing $F_t(X)$. By the definition of F_t, only the legitimate receiver (the one who owns t) will be able to invert $F_t(X)$ to get the message.

Up to now we understand that any public key encryption scheme is based on a difficult problem, and the security thus depends of the associated problem.

Definition 2.4 (Security level)**.** A public key encryption scheme is said to achieve $n-$bit security if an attacker would have to perform 2^n operations to break it or to solve the associated difficult problem.

A very old example of encryption scheme is the McEliece's cryptosystem which consists to use the theory of error-correcting codes to design a one-way function of the form

$$\begin{array}{rccc} F & : & \mathbb{F}_2^k & \longrightarrow & \mathbb{F}_2^n \\ & & \boldsymbol{m} & \longmapsto & \boldsymbol{mG} + \boldsymbol{e} \end{array}$$

where $\boldsymbol{G}$ belongs to $\mathcal{M}_{k,n}(\mathbb{F}_2)$ and $\boldsymbol{e} \in \mathbb{F}_2^n$ is a random binary vector with many zero components. In the following section, we give some backgrounds of coding theory that will allow to understand the McEliece cryptosystem and its variants.

2.2 Error-Correcting Codes

The theory of error-correcting codes have been developed in the second half of the twentieth century, following Shannon's work in 1949. The goal is then to establish clear communications (without noise and without interference). Rather than trying to improve physically the transmission systems, Shannon had the idea of a different approach: Subjecting the signal to a computer processing after receipt in order to detect and correct transmission errors (see figure 2.2).

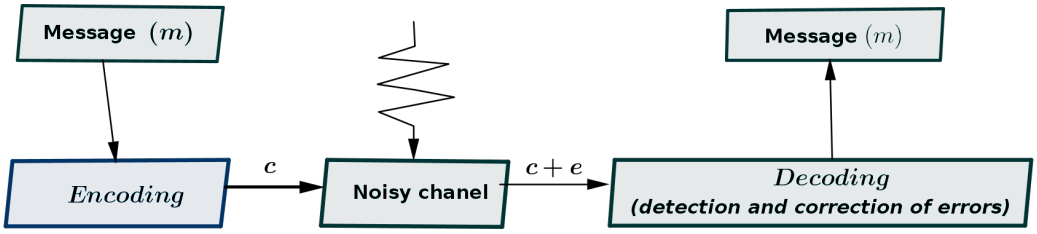

Figure 2.2 – Communication Channel

So, we are interested in the transmission of messages which are each transmitted by a succession of signals. Each message can be written using a $n-$tuple $(x_1, ..., x_n)$, x_i $i = 1, ..., n$ belonging to a set A called the alphabet. The elements of A are then the different signals used and each $n-$tuple obtained is called a "word". The integer n is the length of the words and the set of all words obtained is called a "code". In order to have a code with specific algebraic and combinatorial structures, the alphabet A can be chosen appropriately. Thus the alphabet will be generally a finite field $\mathbb{F}_q$. That is to say a finite field with q elements where q is a power of a prime integer p.

Definition 2.5 (Code)**.** Let $\mathbb{F}_q$ be a finite field with q elements. A code of length n over $\mathbb{F}_q$ is a subset $\mathscr{C}$ of $\mathbb{F}_q^n$.

In order to measure the quantity of transmission errors introduced, we have to use a distance.

Hamming Distance

Let A be a finite alphabet, n a non-zero integer and $d : A^n \times A^n \longrightarrow \mathbb{N}$ the function defined by:

$$d(\boldsymbol{x}, \boldsymbol{y}) = \#\{i \in \{1, ..., n\} : \ x_i \neq y_i\}$$

with $\boldsymbol{x} = (x_1, ..., x_n)$ and $\boldsymbol{y} = (y_1, ..., y_n)$ belonging to A^n.

Proposition 2.1. *The function d is a distance on A^n.*

Proof. Let $\boldsymbol{x} = (x_1, ..., x_n)$, $\boldsymbol{y} = (y_1, ..., y_n)$ and $\boldsymbol{z} = (z_1, ..., z_n)$ belonging to A^n.

- Let's suppose that $d(\boldsymbol{x}, \boldsymbol{y}) = 0$. This means that for all $i \in \{1, ..., n\}$, $x_i = y_i$ and thus $\boldsymbol{x} = \boldsymbol{y}$.
- From the definition of d, it is clear that $d(\boldsymbol{x}, \boldsymbol{y}) = d(\boldsymbol{y}, \boldsymbol{x})$.
- Let $E = \{i \in \{1, ..., n\} : \ x_i \neq y_i\}$, $F = \{i \in \{1, ..., n\} : \ x_i \neq z_i\}$ and $G = \{i \in \{1, ..., n\} : \ z_i \neq y_i\}$. We have $d(\boldsymbol{x}, \boldsymbol{y}) = \#E$, $d(\boldsymbol{x}, \boldsymbol{z}) = \#F$ and $d(\boldsymbol{z}, \boldsymbol{y}) = \#E$. Let i belonging to $\{1, ..., n\}$ and not to $F \cup G$; we have $x_i = z_i$ and $z_i = y_i$. So $x_i = y_i$ and then $i \notin E$; that is to say that $E \subset F \cup G$. This implies that $\#E \leqslant \#(F \cup G) \leqslant \#F + \#G$ and then $d(\boldsymbol{x}, \boldsymbol{y}) \leqslant d(\boldsymbol{x}, \boldsymbol{z}) + d(\boldsymbol{z}, \boldsymbol{y})$.

□

The distance d defined above is the most used in coding theory and is called the "Hamming distance" thanks to Richard Hamming who introduced it in 1950. From this distance, the weight of a vector is defined by:

Definition 2.6 (Hamming weight)**.** The Hamming weight of a word $\boldsymbol{x} \in \mathscr{C}$ denoted by $w(\boldsymbol{x})$ or $w_H(\boldsymbol{x})$ is the distance between $\boldsymbol{x}$ and the zero word.

Different codes have different properties. One of the most important property of a code is its minimum distance, which provides its theoretical error correction capability.

Definition 2.7. Let $\mathscr{C}$ be a code over $\mathbb{F}_q$. The *minimum distance* d of $\mathscr{C}$ is given by:

$$d = \min\{d(\boldsymbol{x}, \boldsymbol{y}) : \ \boldsymbol{x}, \boldsymbol{y} \in \mathscr{C}, \boldsymbol{x} \neq \boldsymbol{y}\}$$

and the *packing radius* is given by

$$t = \lfloor \frac{d-1}{2} \rfloor$$

Decoding

When a word $\boldsymbol{c}$ passes through a transmission channel, generally there are some perturbations and the received word is then $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$, where $\boldsymbol{e}$ is the error introduced. The recipient's objective is to decode $\boldsymbol{y}$; that is to say, find $\boldsymbol{e}$ from $\boldsymbol{y}$ so as to recover the original codeword $\boldsymbol{c}$ or directly the message (see Figure 2.2). In the sequel, we will say that a received word $\boldsymbol{y} = \boldsymbol{c} + \boldsymbol{e}$ contains t errors if $w(\boldsymbol{e}) = t$.

Proposition 2.2 (Unique decoding)**.** *Let $\mathscr{C}$ be a code of length n defined on $\mathbb{F}_q$ with a* packing radius *t. For $\boldsymbol{y} \in \mathbb{F}_q^n$, there exists at most one codeword $\boldsymbol{c} \in \mathscr{C}$ such that $d(\boldsymbol{y}, \boldsymbol{c}) \leqslant t$.*

Proof. Let $\boldsymbol{y} \in \mathbb{F}_q^n$ and suppose that there exist $\boldsymbol{c}$ and $\boldsymbol{c}'$ belonging to $\mathscr{C}$ such that $d(\boldsymbol{y}, \boldsymbol{c}) \leqslant t$ and $d(\boldsymbol{y}, \boldsymbol{c}') \leqslant t$. We then have $d(\boldsymbol{y}, \boldsymbol{c}) + d(\boldsymbol{y}, \boldsymbol{c}') \leqslant 2t \leqslant d - 1$ and thus $d(\boldsymbol{c}, \boldsymbol{c}') \leqslant d - 1 < d$. Since d is the minimum distance, we deduce that $\boldsymbol{c} = \boldsymbol{c}'$. □

Graphically, the decoding of a word $\boldsymbol{y}$ in $\mathbb{F}_q^n$ with a code $\mathscr{C}$ consists to find a codeword $\boldsymbol{c}$ in $\mathscr{C}$ that is closest to $\boldsymbol{y}$. Proposition 2.2 thus state that for $\boldsymbol{y}$ in a ball of radius t centred on a codeword $\boldsymbol{c}$, $\boldsymbol{c}$ is the unique solution of the decoding. In other word, the balls of radius t centred on the codewords do not intersect (see Figure 2.3).

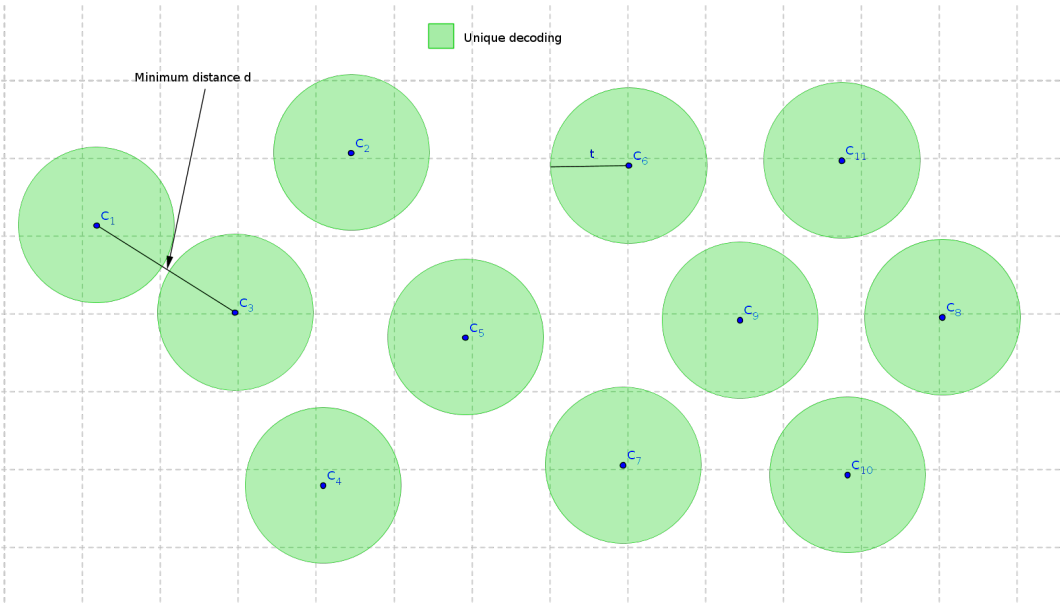

Figure 2.3 – Unique decoding

2.2.1 Linear Codes

Let $\mathbb{F}_q$ be the finite field of q elements, n and k be two non-zero integers. Informally, a linear code $\mathscr{C}$ defined on $\mathbb{F}_q$ is a code that satisfies the property: For all $\boldsymbol{x}$, $\boldsymbol{y}$ in $\mathscr{C}$, and for all α belonging to $\mathbb{F}_q$, $\boldsymbol{x} + \alpha\boldsymbol{y}$ belongs to $\mathscr{C}$.

Definition 2.8 (Linear code)**.** An $(n, k)-$linear code of length n and dimension k on $\mathbb{F}_q$ is a vector subspace of $\mathbb{F}_q^n$ of dimension k.

In the sequel, an $(n, k)-$code will denote an $(n, k)-$linear code and we will say an $(n, k, d)-$ code to denote an $(n, k)-$linear code with minimum distance d.

Remark 2.1. If $\mathscr{C}$ is a linear code then, the minimum distance is the minimum weight of the non-zero codewords.

From the definition, it is clear that linear codes can be defined and represented by matrices.

Definition 2.9 (*Generator matrix*)**.** Let $\mathscr{C}$ be an $(n, k)-$linear code on $\mathbb{F}_q$. A matrix $\boldsymbol{G} \in \mathcal{M}_{k,n}(\mathbb{F}_q)$ is a *generator matrix* of $\mathscr{C}$ if its rows form a basis of $\mathscr{C}$. That is to say:

$$\mathscr{C} = \{\boldsymbol{m}\boldsymbol{G},\ \boldsymbol{m} \in \mathbb{F}_q^k\}$$

Remark 2.2. If $\boldsymbol{G}$ is a *generator matrix* of an $(n,k)-$code $\mathscr{C}$ on $\mathbb{F}_q$ and $\boldsymbol{S}$ in $\mathsf{GL}_k(\mathbb{F}_q)$ then, $\boldsymbol{SG}$ is also a *generator matrix* of $\mathscr{C}$.

Before defining the *parity-check matrix* of a code, we introduce the *dual* of a code.

Definition 2.10. Let $\mathscr{C}$ be an $(n,k)-$code on $\mathbb{F}_q$. The *dual* of $\mathscr{C}$ is the $(n, n-k)-$code $\mathscr{C}^{\perp}$ defined by:

$$\mathscr{C}^{\perp} = \{\boldsymbol{y} \in \mathbb{F}_q^n : \ \boldsymbol{x}\boldsymbol{y}^T = 0 \text{ for all } \boldsymbol{x} \in \mathscr{C}\}$$

Definition 2.11 (*Parity-check matrix*)**.** A *parity-check matrix* of a code $\mathscr{C}$ is a generator matrix of its *dual*.

Remark 2.3. Let $\boldsymbol{G}$ be a generator matrix of an $(n,k)-$code $\mathscr{C}$ over $\mathbb{F}_q$ and $\boldsymbol{H}$ a parity check matrix of $\mathscr{C}$ then we have: $\boldsymbol{G}\boldsymbol{H}^T = \boldsymbol{0}$. Conversely, any matrix $\boldsymbol{H}$ belonging to $\mathcal{M}_{n-k,n}(\mathbb{F}_q)$ of rank $n-k$ that satisfies $\boldsymbol{G}\boldsymbol{H}^T = \boldsymbol{0}$ is a parity-check matrix of $\mathscr{C}$.

Example 2.1. Consider the following matrix $\boldsymbol{G}$ with coefficients in $\mathbb{F}_2$:

$$\boldsymbol{G} = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}$$

The rank of $\boldsymbol{G}$ over $\mathbb{F}_2$ is 4. We can say that $\boldsymbol{G}$ is a generator matrix of an $(7,4)-$code $\mathscr{C}$ over $\mathbb{F}_2$. Let $\boldsymbol{H}$ be the matrix given by:

$$\boldsymbol{H} = \begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

We have $\boldsymbol{H}\boldsymbol{G}^T = \boldsymbol{0}$ and $\mathtt{rank}(\boldsymbol{H}) = 3 = 7-4$, so $\boldsymbol{H}$ is a parity check matrix of $\mathscr{C}$.

From the previous remark, a parity-check (generator) matrix can be computed from a generator (parity-check) matrix in polynomial time. One can also remark that a parity check matrix of a code $\mathscr{C}$ is very helpful to know if a random word $\boldsymbol{y}$ belongs to $\mathscr{C}$ or not. This is an important step during the decoding process and it is achieved by computing the *syndrome*.

Definition 2.12 (*Syndrome*)**.** Let $\boldsymbol{H}$ be a parity check matrix of an $(n,k)-$code $\mathscr{C}$ on $\mathbb{F}_q$ and $\boldsymbol{y}$ belonging to $\mathbb{F}_q^n$. The *syndrome* $s \in \mathbb{F}_q^{n-k}$ of $\boldsymbol{y}$ associated to $\mathscr{C}$ is given by: $s^T = \boldsymbol{H}\boldsymbol{y}^T$

Remark 2.4. A word $\boldsymbol{y} \in \mathbb{F}_q^n$ belongs to a code $\mathscr{C}$ if and only if its *syndrome* associated to $\mathscr{C}$ is equal to $\boldsymbol{0}$.

There is also a relationship between a parity check matrix and the *minimum distance* of the associated code, that is given by the following proposition.

Proposition 2.3. *Let $\mathscr{C}$ be an $(n,k)-$code with $\boldsymbol{H}$ as a parity check matrix. The minimum distance of $\mathscr{C}$ is d if and only if d is the biggest integer such that any sub-matrix constituted by $d-1$ columns of $\boldsymbol{H}$ is of rank $d-1$.*

Proof. The proof comes from the fact that a word $\boldsymbol{y} \in \mathbb{F}_q^n$ belongs to $\mathscr{C}$ if and only if $\boldsymbol{H}\boldsymbol{y}^T = \boldsymbol{0}$. □

From the previous proposition we can give a proof of the singleton bound.

Theorem 2.4 (*Singleton bound*)**.** *If $\mathscr{C}$ is a $(n,k,d)-$code then $d \leqslant n-k+1$.*

Proof. Let $\boldsymbol{H} = (\boldsymbol{u}_1, ..., \boldsymbol{u}_n)$ be a parity check matrix of $\mathscr{C}$ with $\boldsymbol{u}_i^t \in \mathbb{F}_q^{n-k}$ $(i = 1, ..., n)$. We recall that, since the rank of $\boldsymbol{H}$ is $n-k$, the matrix $(\boldsymbol{u}_1, ..., \boldsymbol{u}_{d-1})$ is of rank at most $n-k$. So if $d-1 > n-k$ then the rank of the matrix $(\boldsymbol{u}_1, ..., \boldsymbol{u}_{d-1})$ is not $d-1$. From proposition 2.3, this contradicts the fact that d is the minimum distance of $\mathscr{C}$. □

Theorem 2.4 gives an upper bound of the minimum distance of a code. For fixed values n and k, we want a code with a d as large as possible since such a code can intrinsically correct more errors. The more d nears $n-k+1$, the more the code is optimal (i.e., may correct more errors).

Definition 2.13 (MDS Code)**.** An $(n,k,d)-$code $\mathscr{C}$ is said to be MDS (Maximum Distance Separable) if the singleton bound is reached. That is to say:

$$d = n - k + 1$$

The optimality of a code is an important parameter of efficiency in terms of decoding, but it's not the only one. A high minimum distance ensures that the code can theoretically correct many errors, but this does not guarantee the existence of an efficient decoding algorithm.

2.2.2 The general decoding problem

The general decoding problem is at the base of several cryptosystems based on coding theory. The term *"general"* here denotes the fact that there is no information about the structure of the given linear code. The problem can be described as follows:

Problem 2.5. *Let $\mathscr{C}$ be an $(n,k,d)-$code on $\mathbb{F}_q$, $\boldsymbol{y} \in \mathbb{F}_q^n$ and t a given integer. Find $\boldsymbol{c}$ in $\mathscr{C}$ such that*

$$d(\boldsymbol{y}, \boldsymbol{c}) \leqslant t$$

This problem also termed as *Bounded distance decoding problem* was first studied in [BMvT78] and was proven to be NP-hard. The problem can be described by using a generator matrix $\boldsymbol{G}$ of the code $\mathscr{C}$ as follows:

Problem 2.6 (General Decoding Problem)**.** *Let* $\boldsymbol{G}$ *be a full-rank matrix belonging to* $\mathcal{M}_{k,n}(\mathbb{F}_q)$ *with* $k \leqslant n$, $\boldsymbol{y}$ *an element of* $\mathbb{F}_q^n$ *and* t *an integer. The* General Decoding Problem $\mathsf{GD}_{n,k,t}$ *is to find* $\boldsymbol{e}$ *in* $\mathbb{F}_q^n$ *and* $\boldsymbol{m}$ *in* $\mathbb{F}_q^k$ *such that* $\boldsymbol{y} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{e}$ *with* $w(\boldsymbol{e}) \leqslant t$.

Or by using a parity check matrix $\boldsymbol{H}$ of $\mathscr{C}$ as follows:

Problem 2.7 (Syndrome Decoding Problem)**.** *Let* $\boldsymbol{H}$ *be a full-rank matrix belonging to* $\mathcal{M}_{n-k,n}(\mathbb{F}_q)$ *with* $k \leqslant n$, $\boldsymbol{s}$ *an element of* $\mathbb{F}_q^{n-k}$ *and* t *an integer. The* Syndrome Decoding Problem *is to find* $\boldsymbol{e}$ *in* $\mathbb{F}_q^n$ *such that* $\boldsymbol{s} = \boldsymbol{e}\boldsymbol{H}^t$ *with* $w(\boldsymbol{e}) \leqslant t$.

The best algorithms that solve this problem are derived from *information set decoding* introduced by Prange in [Pra62]. In its simplest form, the decoder tries to find a subset of k columns of the generator matrix, that is error-free and for which the sub-matrix composed by this subset is invertible. The message can then be recovered by multiplying the corresponding codeword at the right by the inverse of this sub-matrix. The algorithm has been optimized during several years (see [LB88, Leo88, Ste88], [MMT11, BJMM12]) but the best one remains exponential on the length of $\mathscr{C}$ (on average $\mathscr{O}(2^{n/20})$ operations for binary codes.

There are code families for which the later problem is no longer difficult and for which efficient decoding algorithms are known. In the subsection that follows, we recall some of the linear codes that are used for cryptographic purpose.

2.2.3 Examples of decodable families of codes

We have seen in the previous paragraphs that the decoding problem is difficult when dealing with an unknown family of codes. It is not the case with structured codes. We briefly introduce here some families of codes that can be decoded efficiently, that is to say equipped with a polynomial decoding algorithm.

Generalized Reed-Solomon and Goppa codes.

Generalized Reed-Solomon codes, or shortly GRS codes, were introduced by Reed and Solomon in [RS60] and represent a powerful family of codes with many applications. Ten years after, binary Goppa codes were introduced by Valery Goppa [Gop70]. Goppa codes can be defined as subfield subcodes of GRS codes.

Definition 2.14 (Generalized Reed-Solomon codes)**.** Let k and n be two integers such that $1 \leqslant k < n \leqslant q$ where $q = p^m$ is a power of a prime number p. Let $(\boldsymbol{x}, \boldsymbol{y}) \in \mathbb{F}_q^n \times \mathbb{F}_q^n$ be a pair such that $\boldsymbol{x}$ is an n-tuple of distinct elements of $\mathbb{F}_q$ and the elements y_i are non-zero elements in $\mathbb{F}_q$. The Generalized Reed-Solomon code $\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y})$ is given by:

$$\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\text{def}}{=} \{(y_1 f(x_1), \ldots, y_n f(x_n)) \mid f \in \mathbb{F}_q[x]\ ,\ \deg(f) < k\}\ .$$

The vector $\boldsymbol{x}$ is called the support of the code and $\boldsymbol{y}$ the multiplier vector. One can easily deduce that a generator matrix of $\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y})$ is given by

$$\boldsymbol{G} = \begin{pmatrix} 1 & 1 & \dots & 1 \\ x_1 & x_2 & \dots & x_n \\ \vdots & \vdots & \vdots & \vdots \\ x_1^{k-1} & x_2^{k-1} & \dots & x_n^{k-1} \end{pmatrix} \begin{pmatrix} y_1 & & & \\ & y_2 & & 0 \\ & & \ddots & \\ 0 & & & y_n \end{pmatrix}.$$

Proposition 2.8 ([MS86] Theorem 4, Chapter 10)**.** *The dual of a GRS code is also a GRS code and we have*

$$\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y})^{\perp} = \mathbf{GRS}_{n-k}(\boldsymbol{x}, \boldsymbol{z}),$$

where $\boldsymbol{z}$ is a non-zero codeword of the $(n, 1, n)$ GRS code $\mathbf{GRS}_{n-1}(\boldsymbol{x}, \boldsymbol{y})^{\perp}$.

We notice that a vector $\boldsymbol{z}$ with $z_i \neq 0$ (for $i = 1...n$) exists since any non-zero codeword of a $(n, 1, n)-$GRS code has a Hamming weight equal to n. From the propositions 2.8 and 2.3, one can deduce that GRS codes are MDS. They are also known to possess fast decoding algorithms that can correct efficiently up to $\frac{n-k}{2}$ errors (see for example [Gao03] or [MS86] for more details).

Definition 2.15 (Alternant codes)**.** Let r be another non-zero integer. A $p-$ary alternant code of length n is a linear code over $\mathbb{F}_p$ defined from a GRS code $\mathbf{GRS}_r(\boldsymbol{x}, \boldsymbol{y}) \subset \mathbb{F}_{p^m}^n$ by

$$\mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\text{def}}{=} \mathbf{GRS}_r(\boldsymbol{x}, \boldsymbol{y})^{\perp} \cap \mathbb{F}_p^n.$$

Definition 2.16 (Binary Goppa codes)**.** Let $\boldsymbol{x} \in \mathbb{F}_{2^m}^n$ be a $n-$*tuple* of distinct elements and $g \in \mathbb{F}_{2^m}[x]$ be a polynomial of degree t such that $g(x_i) \neq 0$ for all i. The binary Goppa code $\mathscr{G}(\boldsymbol{x}, g)$ is the $2-$ary alternant code $\mathscr{A}_t(\boldsymbol{x}, \boldsymbol{y})$ with $\boldsymbol{y} \stackrel{\text{def}}{=} (1/g(x_1), \dots, 1/g(x_n))$.

As for GRS codes, Goppa codes are known to possess several decoding algorithms that can decode up to t errors in polynomial time [Pat75, MS86, BML13].

Reed-Muller codes.

Reed-Muller codes were introduced by David Muller [Mul54] and rediscovered shortly after with an efficient decoding algorithm by Irving Reed [Ree54].[1] The scientific community was highly interested in this family of codes and therefore discovered many structural properties of Reed-Muller codes. Recently Kudekar, Mondelli, Sasoglu and Urbanke proved that Reed-Muller codes achieve the capacity of the Binary Erasure channel [KKM+17].

[1] Although it seems that these codes were firstly discovered by Mitani in 1951 [Mit51], they became popular only after the article of Muller and Reed.

Definition 2.17 (Reed-Muller code)**.** Let $\mathbb{F}_2[x_1, \ldots, x_m]$ be the set of boolean polynomials with m variables. Let us set $\{a_1, \ldots, a_n\} \stackrel{\text{def}}{=} \mathbb{F}_2^m$ and $n \stackrel{\text{def}}{=} 2^m$. The Reed-Muller code denoted by $\mathcal{RM}(r, m)$ with $0 \leqslant r \leqslant m$ is the linear space defined by:

$$\mathcal{RM}(r, m) \stackrel{\text{def}}{=} \Big\{ \big(f(a_1), \ldots, f(a_n)\big) \ : \ f \in \mathbb{F}_2[x_1, \ldots, x_m], \deg f \leqslant r \Big\}$$

We have the following theorem that gives the dimension of a Reed-Muller code:

Theorem 2.9. *The dimension of* $\mathcal{RM}(r, m)$ *is equal to* $\sum_{i=0}^{r} \binom{m}{i}$.

Proof. For given integers m and r with $0 \leqslant r \leqslant m$, let $\mathbb{F}_2[x_1, \ldots, x_m]^r$ be the set of elements f of $\mathbb{F}_2[x_1, \ldots, x_m]$ that satisfy $\deg f \leqslant r$. One can see that $\mathbb{F}_2[x_1, \ldots, x_m]^r$ is a $\mathbb{F}_2-$vector space generated by the basis

$$B_r = \{x_1^{u_1} \ldots x_m^{u_m} \ : \ u_i \in \{0, 1\}, \ \sum_i u_i \leqslant r\}$$

This implies that $\dim\left(\mathbb{F}_2[x_1, \ldots, x_m]^r\right) = \sum_{i=0}^{r} \binom{m}{i}$. Furthermore, from the definition of $\mathcal{RM}(r, m)$, it is clear that $\mathcal{RM}(r, m)$ is isomorphic to $\mathbb{F}_2[x_1, \ldots, x_m]^r$. Hence $\dim\left(\mathcal{RM}(r, m)\right) = \sum_{i=0}^{r} \binom{m}{i}$. □

Another nice property of the set of Reed-Muller codes is that like GRS codes they are stable by the action of the dual.

Theorem 2.10 ([MS86] Chapter 13.)**.**

$$\mathcal{RM}(r, m)^{\perp} = \mathcal{RM}(m - r - 1, m)$$

LDPC and MDPC codes.

Another important class of linear codes is the family of low density parity check (LDPC) codes discovered by Gallager [Gal63]. He was motivated by the problem of finding "random-like" codes that could be decoded near the channel capacity with quasi-optimal performance and feasible complexity. These codes were extended in a natural way to moderate density parity check codes in [OB09]. LDPC codes have many applications in communication field as well as in cryptography.

Definition 2.18 (LDPC/MDPC codes)**.** An $[n, k, \omega]-$code is a linear code defined by a $k \times n$ parity-check matrix ($k < n$) where each row has weight ω.

An *LDPC* code is an $[n, k, \omega]-$code with $\omega = O(1)$, when $n \to \infty$.[Gal63]

An *MDPC* code is an $[n, k, \omega]-$code with $\omega = O(\sqrt{n})$, when $n \to \infty$.[OB09]

The theory of error correcting codes is not only a highly important tool in the communication field, it is also applied to public key cryptography. One of the oldest public key encryption scheme, namely the McEliece PKC [McE78], is based on several aspects from coding theory.

2.3 Code-Based Public-Key Encryption Schemes

In this section we give the basic notions about the McEliece [McE78] cryptosystem.

2.3.1 McEliece encryption scheme

Let $\mathcal{G}$ be a family of (n, k)-linear codes over $\mathbb{F}_q$ for which a polynomial-time algorithm to decode t-error is available. The general version of the McEliece cryptosystem is described as follows.

Key generation.

1. Let $\boldsymbol{G}' \in \mathcal{M}_{k,n}(\mathbb{F}_q)$, be a generating matrix of a t-error correcting code $\mathscr{C}' \in \mathcal{G}$
2. Pick an $n \times n$ permutation matrix $\boldsymbol{P}$ and a $k \times k$ invertible matrix $\boldsymbol{S}$ at random over $\mathbb{F}_q$.
3. Compute $\boldsymbol{G} = \boldsymbol{S}^{-1}\boldsymbol{G}'\boldsymbol{P}^{-1}$ which is another generating matrix.

The public key is $(\boldsymbol{G}, t)$ and the private key is $(\boldsymbol{S}, \boldsymbol{G}', \boldsymbol{P})$.

Encryption. To encrypt the message $\boldsymbol{m} \in \mathbb{F}_q^k$, one randomly generates $\boldsymbol{e} \in \mathbb{F}_q^n$ of Hamming weight $\leqslant t$. The ciphertext is then the vector $\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{e}$.

Decryption. The vector $\boldsymbol{c}\boldsymbol{P}^{-1}$ is at a distance at most t of $\mathscr{C}$. The decoding algorithm thus allows to find the vector $\boldsymbol{y} \stackrel{\text{def}}{=} \boldsymbol{m}\boldsymbol{S}^{-1}$. The plaintext is deduced by computing $\boldsymbol{y}\boldsymbol{S}$.

A version of the McEliece cryptosystem that uses the parity-check matrix instead of the generating matrix has been proposed by Niederreiter [Nie86], and has been proved to be completely equivalent in term of security [LDW94].

2.3.2 Niederreiter encryption scheme

The Niederreiter cryptosystem is generally describes as follows.

Key generation.

1. Let $\boldsymbol{H}' \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$, be a parity check matrix of a t-error correcting code $\mathscr{C}' \in \mathcal{G}$
2. Pick at random an $n \times n$ permutation matrix $\boldsymbol{P}$ and a $(n-k) \times (n-k)$ non-singular matrix $\boldsymbol{S}$ over $\mathbb{F}_q$.
3. Compute $\boldsymbol{H} = \boldsymbol{S}^{-1}\boldsymbol{H}'\boldsymbol{P}^{-1}$.

The public key is $(\boldsymbol{H}, t)$ and the private key is $(\boldsymbol{S}, \boldsymbol{H}', \boldsymbol{P})$.

Encryption. For a message $\boldsymbol{m} \in \mathbb{F}_q^n$ of Hamming weight $\leqslant t$. The cipher text is given by $\boldsymbol{c} = \boldsymbol{H}\boldsymbol{m}^T$.

Decryption. Since $\boldsymbol{c} = \boldsymbol{S}^{-1}\boldsymbol{H}'\boldsymbol{P}^{-1}\boldsymbol{m}^T = \boldsymbol{S}^{-1}\boldsymbol{H}'(\boldsymbol{m}\boldsymbol{P})^T$ and $\boldsymbol{m}\boldsymbol{P}$ is a word of weight less than or equal to t, the receiver decodes $\boldsymbol{S}\boldsymbol{c}$ to get the word $\boldsymbol{y}$. The associated plaintext is then $\boldsymbol{y}\boldsymbol{P}$.

2.3.3 Security of the system

The security of the McEliece cryptosystem is based on two facts: firstly the public code is supposed to be indistinguishable from a random code. If the later supposition is satisfied then in order to decrypt a cyphertext, one has to solve an instance of Problem 2.7 (Syndrome decoding problem) which is known as a difficult problem. In the usual security framework there are three levels of attacks that might jeopardize the scheme.

- ⋆ Distinguishing Attacks: an attacker has to distinguish between the public code and a random code in order to invalidate the hypothesis of the security proofs for the scheme. In some cases the distinguisher might lead to an efficient Key Recovery Attack.

- ⋆ Message Recovery Attacks (MRA): an attacker tries to retrieve the message from a given ciphertext.

- ⋆ Key Recovery Attacks (KRA): an adversary tries to retrieve the private key from the public key and thus completely breaks the cryptosystem.

So the security clearly depends of the family of codes used, and the chosen parameters. In his original paper, McEliece proposed to use a $(1024, 524, 101)-$binary Goppa code. Thus its security is based on two problems:

- The difficulty of decoding a random linear code

- The difficulty of recovering a decoding algorithm from a public matrix representation of a binary Goppa code.

Although the second problem has never been proven NP-hard, the system has withstood all structural attacks until today. A **distinguisher** exists in the case of high rate Goppa codes [FGO+13] but, despite of this potential vulnerability, there is no efficient algorithm for the moment exploiting the knowledge and the properties of the distinguisher. The system is not used in practice because of the enormous size of the public key. Table 2.1 gives some updated parameters for an acceptable security level and compares the associated public key sizes to RSA public key sizes (see [BLP08, NMBB10] for more details).

Security level	(n, k)	t	Public Key size	RSA Public key sizes
80 bits	$(1632, 1269)$	33	460647 bits	512 bits
128 bits	$(2960, 2288)$	56	1537536 bits	3072 bits
256 bits	$(6624, 5129)$	115	7667855 bits	15360 bits

Table 2.1 – Parameters and key size for McEliece with Goppa codes from [BLP08] and key size for the RSA scheme

One can remark that for the same level of security, the public key of the McEliece cryptosystem is more than 100 times greater than the RSA public key [RSA78]. Therefore reducing the size of the keys is one of the starting points of a continuous research interest in this field. We mention the existence of a recent compact variant of the McEliece scheme based on quasi-dyadic Goppa codes due to Misoczki and Barreto [MB09], variant that is not yet broken in the binary case. However, there are several ideas for solving this problem of key sizes. The very old one consists to replace the Goppa codes by another family of codes.

2.3.4 Some variants of the McEliece cryptosystem

The natural question with the McEliece cryptosystem is always to know the most appropriate code family to use.

Generalized Reed-Solomon codes

This family was proposed for the first time by Niederreiter in [Nie86] but turned out to be an insecure solution. Indeed, six years after the article was published, Sidelnikov and Shestakov proposed a polynomial time attack against this variant [SS92]. Nevertheless the idea of using GRS codes was reconsidered more than ten years after by Berger and Loidreau when they proposed to consider subcodes of GRS codes [BL05]. Unfortunately this technique was also attacked in two steps by Wieschebrink [Wie06a, Wie09], using the **square code structure**.

Other attempts to repair the Niederreiter variant were proposed by Wieschebrink [Wie06b] whose idea was to add random columns to the generator matrix. But this variant turned out to be extremely unsecure against **square code type attacks** or **filtration type attacks** [CGG+14]. Nevertheless GRS codes are still of high interest for cryptography since several modified version of the McEliece scheme use this family of codes. For example Baldi et al. [BBC+16] proposed to change the permutation matrix, Tillich et al. [CT16] proposed to use them in a "$u \mid u + v$" construction, Wang [Wan16] proposes to use a more general technique derived from Wieschebrink's idea.

Reed-Muller codes

Reed-Muller codes were proposed by Sidelnikov's in [Sid94] and was firstly attacked by Minder and Shokrollahi [MS07]. In the case of Reed-Muller codes, the Key Recovery Attack is reduced to solving the code equivalence problem:

Problem 2.11 (Permutation Code Equivalence Problem)**.** *Let* $\boldsymbol{G}$ *and* $\boldsymbol{G}^*$ *be the generating matrices for two* $[n, k]$ *binary linear codes. Given* $\boldsymbol{G}$ *and* $\boldsymbol{G}^*$*, find a* $k \times k$ *invertible matrix* $\boldsymbol{S}$ *and* $n \times n$ *permutation matrix* $\boldsymbol{P}$ *such that* $\boldsymbol{G}^* = \boldsymbol{SGP}$*.*

Since there is only one Reed-Muller code with parameters r and m, a cryptanalysis can try to solve problem 2.11 with $\boldsymbol{G}^* = \boldsymbol{G}_{\text{pub}}$ and $\boldsymbol{G}$ a generator matrix of $\mathcal{RM}(r, m)$. Minder and Shokrollahi managed to solve this problem using a filtration type attack based on the structure properties of the minimum weight codewords. The complexity of their algorithm was dominated by the minimum weight codewords searching algorithm.

Recently, Chizhov and Borodin [CB14] proposed another attack that could solve the code equivalence problem, for some of the parameters of the Reed-Muller codes, in polynomial time. Their idea was to use two simple operations in order to find the first order Reed-Muller code given the r^{th} order Reed-Muller code. Indeed they noticed that the dual and the **square code** of a Reed-Muller code is still a Reed-Muller code. So they combined these operations in order to approach the $\mathcal{RM}(1, m)$. A modified version using the masking technique introduced by Wieschebrink was proposed in [GM13] but we will prove in Chapter 3, using a **square code** type attack, that this variant is insecure.

Algebraic-geometry codes

This family of codes was suggested by Janwa and Moreno [JM96]. Several articles discuss the potential vulnerabilities of this variant and propose algorithms that could be deployed to attack in some particular cases (codes from curves of genus at most 2) [Min07, FM08]. Nevertheless they can not be generalized and suffer in terms of efficiency. In [CMCP14] Couvreur, Marquez-Corbella and Pellikaan proposed a polynomial type algorithm that works on codes from curves of arbitrary genus.

LDPC codes

Monico, Rosenthal and Shokrollahi were the first ones to propose and analyze a McEliece variant using low density parity check codes in [MRAS00]. Using the idea of Gaborit to consider quasi-cyclic codes [Gab05] [2] the new QC-LDPC cryptosystem was presented by Baldi and Chiaraluce in [BC07]. Both BCH codes and LDPC codes with quasi-cyclic structure were successfully cryptanalyzed by Otmani, Tillich and Dallot [OTD08]. In order to prevent the last attack, a modification based on

[2] In [Gab05] the author proposes BCH codes with quasi-cyclic structure. The idea of adding the quasi cyclic structure became one of the main techniques for reducing the key size in the McEliece scheme.

increasing the weight of the codewords in the public code was proposed in [BBC08]. More details about this variant can be found in [Bal14]. The modification of [BBC08] seems to be working for the moment since there is no other structural attacks.

MDPC codes

Moderate Density Parity-Check codes are probably the most suitable codes in a McEliece type scheme [MTSB13]. Many cryptographic arguments are in favour of this family of codes like efficiency, small key size when used with a quasi-cyclic structure and the most important to our opinion the lack of algebraic structure. Another security argument is the fact that the usual distinguisher does not work for MDPC codes. In a recent paper, weak keys of the QC-MDPC scheme are revealed [BDLO16]. However the authors show how to avoid vulnerable parameters.

Wild Goppa codes

This code family is a natural extension from binary Goppa codes to non-binary fields. It was proposed by Bernstein, Lange and Peters in [BLP10] and [BLP11]. Many of the proposed parameters were broken by Couvreur, Otmani and Tillich using **filtration type techniques** when the extension is quadratic [COT14a, COT14b].

Srivastava codes

Srivastava codes were proposed in [Per12] in order to reduce the key length of the original McEliece scheme. The author uses Quasi-Dyadic Srivastava codes and gives another application of these types of codes for signature schemes. Even though the parameters for the signature were broken in [FOP+16a], the parameters for the encryption scheme are still valid.

Convolutional codes

Convolutional codes represented among the shortest term solutions since between the proposed article by Londahl and Johansson [LJ12] and the efficient attack by Landais and Tillich [LT13] only one year passed.

Polar codes

The first variant using Polar codes was proposed by Shrestha and Kim [SK14] while the second one using subcodes of Polar codes was given in [HSEA14]. In [BCD+16] the first variant was attacked using the structure of the minimum weight codewords. The authors managed to solve the code equivalence problem for Polar codes and thus completely break the scheme.

To conclude this chapter, we emphasise that there are code families which are not appropriate in this context due to their structural properties, namely the GRS codes, the Reed-Muller codes, the Polar codes ... However several classes of codes

remain secure in a McEliece PKC such as original binary Goppa codes and MDPC codes etc. We also point out that all these variants use hamming distance. Another idea that is very similar to all the above variants is to use another family of codes, but with another metric. Gabidulin was the first to introduce this idea with the GPT cryptosystem [GPT91] that uses the rank distance. After the attacks of [Ove05b] several variants of the system were proposed [Gab08, GRH09, RGH10, RGH11]. The chapters 4 and 5 of this thesis will be devoted to this part of code based cryptography which is nowadays known as rank-based cryptography. We will show that all existing variants of the GPT cryptosystem are insecure.

Chapter 3

Cryptanalysis of a Modified Sidelnikov Cryptosystem

Introduction

This chapter develops a cryptanalysis of the modified version given in [GM13] of the Sidelnikov encryption scheme [Sid94] which is a McEliece-type public key encryption scheme [McE78] based on Reed-Muller codes. The idea of [GM13] is to add random columns to prevent the key-recovery attacks of [MS07, CB13]. But, like Reed-Solomon codes, Reed-Muller codes are evaluation codes and because of this, they can be distinguished from random codes. These two families of codes share very similar properties which facilitates the recovering of the random columns. Our key-recovery attack is divided into two steps. The first one is an adaptation to Reed-Muller codes of the attacks presented in [GOT12b, CGG$^+$14] in order to find the secret random columns. This is achieved in $O(n^5)$ operations in the binary field where n is the block length of the codes. The second step applies [MS07, CB13] to recover the secret permutation that hides the structure of the Reed-Muller codes. The rest of the chapter is devoted to the description of the first step of the attack.

3.1 Preliminary Facts

We give here some definitions and properties from coding theory we need in the chapter. Let $\mathbb{F}_q$ be the finite field of q elements, n and k be two non-zero integers such that $k \leqslant n$.

Definition 3.1. Let $\mathscr{C}$ be a $(n, k)-$code over $\mathbb{F}_q$ and i in $\{1, \cdots, n\}$. The punctured code $\mathscr{C}^i$ of $\mathscr{C}$ is obtained by puncturing (or deleting) the $i-$th coordinate from all the codewords of $\mathscr{C}$.

Definition 3.2 (Component-wise product)**.** Given two vectors $\boldsymbol{a} = (a_1, \ldots, a_n)$ and $\boldsymbol{b} = (b_1, \ldots, b_n)$ in $\mathbb{F}^n$ where $\mathbb{F}$ is field, we denote by $\boldsymbol{a} \star \boldsymbol{b}$ the component-wise product:

$$\boldsymbol{a} \star \boldsymbol{b} \stackrel{\text{def}}{=} (a_1 b_1, \ldots, a_n b_n).$$

Definition 3.3 (Product of codes)**.** Let $\mathscr{A}$ and $\mathscr{B}$ be two linear codes of length n. The star product code denoted by $\mathscr{A} \star \mathscr{B}$ of $\mathscr{A}$ and $\mathscr{B}$ is the vector space spanned by all products $a \star b$ where a and b range over A and B respectively.

When $\mathscr{B} = \mathscr{A}$ then $\mathscr{A} \star \mathscr{A}$ is called the square code of $\mathscr{A}$ and is rather denoted by $\mathscr{A}^2$.

Let S_n be the permutations group of order n and σ belonging to S_n. From the above definitions, the following corollary is obvious:

Corollary 3.1. *For any linear code $\mathscr{A}$ of length n,*

$$(\mathscr{A}^\sigma)^2 = (\mathscr{A}^2)^\sigma$$

The importance of the square code construction becomes clear when we compare the dimensions of a code $\mathscr{A}$ with the dimension of its square code $\mathscr{A}^2$ and one major question is to know what one should expect. This comparison has already been made in [GOT12b, CGG+14] in the case of generalized Reed-Solomon codes which allowed to mount efficient attacks on several different schemes based on generalised Reed-Solomon codes [Wie09, GOT12b, CGG+14]. The results of this chapter are based on these comparisons in the case of Reed-Muller codes.

We recall now important facts about the dimension of product of codes.

Proposition 3.2. *For any linear subspaces $F \subseteq E$ and $G \subseteq E$ with finite dimensions:*

$$\dim F \star G \leqslant \dim F \dim G - \binom{\dim F \cap G}{2}. \tag{3.1}$$

Proof. Assume $d \stackrel{\text{def}}{=} \dim F \cap G$ and let $\mathcal{B} = \{b_1, \ldots, b_d\}$ be a basis of $F \cap G$. We complete $\mathcal{B}$ with vectors $\mathcal{F} = \{f_1, \ldots, f_t\}$ so that $\mathcal{B} \cup \mathcal{F}$ is a basis of F. We do the same for G by completing $\mathcal{B}$ with $\mathcal{G} = \{g_1, \ldots, g_m\}$ so that $\mathcal{B} \cup \mathcal{G}$ is a basis of G. A generating set of $F \star G$ is the union of the four sets $\{b_i \star b_j : 1 \leqslant i \leqslant j \leqslant d\}$, $\{b_i \star f_j : 1 \leqslant i \leqslant d, 1 \leqslant j \leqslant t\}$, $\{b_i \star g_j : 1 \leqslant j \leqslant d, 1 \leqslant j \leqslant m\}$ and $\{f_i \star g_j : 1 \leqslant j \leqslant t, 1 \leqslant j \leqslant m\}$. The proof is terminated by observing the equality:

$$dt + dm + tm + \binom{d+1}{2} = (t+d)(d+m) - \frac{1}{2}d(d-1).$$

□

Corollary 3.3. *For any linear subspaces $F \subseteq E$:*

$$\dim F \star E \leqslant \dim F \dim E - \binom{\dim F}{2}.$$

In particular

$$\dim E^2 \leqslant \binom{\dim E + 1}{2} \tag{3.2}$$

In practice, the upper bound of (3.2) is generally reach with a high probability. That means $\dim E^2 = \binom{\dim E+1}{2}$. For more details, see [GOT12b, CGG+14].

3.2 Wieschebrink's Masking Technique

Here we present a masking technique first developed in [Wie06b] and then proposed several times with different families of codes. It consists in inserting random columns in the secret matrix. This technique can be used both in the McEliece cryptosystem and the Niederreiter version.

3.2.1 Modified McEliece scheme

Key generation.

1. Choose three integers n_0, k, ℓ with $\ell \ll n_0$ and set $n \stackrel{\text{def}}{=} n_0 + \ell$. Pick at random a generating matrix $\boldsymbol{G}_0$ of an $(n_0, k)-$code $\mathscr{C}$ that is able to decode t errors.

2. Pick randomly a matrix $\boldsymbol{R}$ in $\mathcal{M}_{k,\ell}(\mathbb{F}_q)$, an invertible matrix $\boldsymbol{S}$ in $\mathsf{GL}_k(\mathbb{F}_q)$ and a $n \times n$ permutation matrix $\boldsymbol{P}$.

3. Set $\boldsymbol{G}' = (\boldsymbol{G}_0 \mid \boldsymbol{R})$ and compute $\boldsymbol{G} = \boldsymbol{S}^{-1}\boldsymbol{G}'\boldsymbol{P}^{-1}$.

The public key is $(\boldsymbol{G}, t)$ and the private key is $(\boldsymbol{S}, \boldsymbol{P}, \boldsymbol{G}')$.

Encryption. To encrypt a plaintext $\boldsymbol{m} \in \mathbb{F}_q^k$, one randomly generates $\boldsymbol{e} \in \mathbb{F}_q^n$ of weight less than t and computes the ciphertext $\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{e}$.

Decryption. To decrypt $\boldsymbol{c}$, one computes $\boldsymbol{y} = \boldsymbol{c}\boldsymbol{P}$ and let $\boldsymbol{y}'$ be the n_0 first columns of $\boldsymbol{y}$. The vector $\boldsymbol{y}'$ is located within distance t from $\mathscr{C}$. The decoding of $\boldsymbol{y}'$ provides the plaintext.

3.2.2 Modified Niederreiter scheme

Here one can apply the same principle as in the case of McEliece cryptosystem, but here the insertion of random columns is done in the parity check matrix.

Key generation.

1. Choose three integers n_0, k, t, ℓ with $\ell \ll n_0$ and set $n \stackrel{\text{def}}{=} n_0 + \ell$. Pick a random parity-check matrix $\boldsymbol{H}_0$ of an $(n_0, k)-$code $\mathscr{C}$ that is able to decode t errors.

2. Pick randomly a matrix $\boldsymbol{R}$ in $\mathcal{M}_{(n_0-k),\ell}(\mathbb{F}_q)$, a non singular matrix $\boldsymbol{S}$ in $\mathsf{GL}_{n_0-k}(\mathbb{F}_q)$ and a $n \times n$ permutation matrix $\boldsymbol{P}$.

3. Set $\boldsymbol{H}' = (\boldsymbol{H}_0 \mid \boldsymbol{R})$ and compute $\boldsymbol{H} = \boldsymbol{S}^{-1}\boldsymbol{H}'\boldsymbol{P}^{-1}$

The public key is $(\boldsymbol{H}, t)$ and the private key is $(\boldsymbol{S}, \boldsymbol{H}', \boldsymbol{P})$.

Encryption. For a plaintext $\boldsymbol{m} \in \mathbb{F}_q^n$ of Hamming weight less than t, the corresponding ciphertext is given by $\boldsymbol{c} = \boldsymbol{H}\boldsymbol{m}^T$.

Decryption. Let $\mathsf{dec}(\cdot)$ be the decoding algorithm of $\mathscr{C}$. The symbol $\perp$ stands for a decoding failure[1]. The decryption of a ciphertext $\boldsymbol{c}$ is described in Algorithm 1.

Algorithm 1 Decryption of Niederreiter scheme with Wieschebrink's masking.

$\boldsymbol{u} = \perp$
for all $\boldsymbol{z} \in \mathbb{F}_q^\ell$ **do**
  $\boldsymbol{y} = \mathsf{dec}\left(\boldsymbol{S}\boldsymbol{c} + \boldsymbol{R}\boldsymbol{z}^T\right)$
  if $\boldsymbol{y} \neq \perp$ **then**
    $\boldsymbol{u} = (\boldsymbol{y}, \boldsymbol{z})\boldsymbol{P}$, **return** $\boldsymbol{u}$
  end if
end for
return $\boldsymbol{u}$

Note that it is possible for the word $\boldsymbol{u}$ to be different from the transmitted message $\boldsymbol{m}$. But an analysis of the meaning of the received message can eliminate these cases and consider them as failures decoding. The complexity of this algorithm is of order $q^\ell T(\mathsf{dec})$ where $T(\mathsf{dec})$ is the time complexity of the decoding algorithm $\mathsf{dec}(\cdot)$.

Although the public code seems to be random in this description, a major problem rests on the choice of the code family to use and how to reduce the size of the keys. Wieschebrink had proposed the use of Reed-Solomon codes but in [GOT12b, CGG+14] an attack is presented that can recover the random secret matrix $\boldsymbol{R}$. Recently, the paper [GM13] suggested the use of Reed-Muller codes along with Wieschebrink's masking technique to propose a McEliece-type encryption scheme. In the next section, we describe how to find the random columns of $\boldsymbol{R}$ in this case. Our attack uses the same technique as the one presented in [GOT12b, CGG+14] for the case of Reed-Solomon codes.

3.3 Recovering the Random Columns in Polynomial Time

In this section, we draw inspiration from [GOT12b, CGG+14] to mount an attack on the version presented in [GM13]. But before doing so, we present some properties of Reed-Muller codes.

3.3.1 Some Properties of Reed-Muller Codes

Let's start by the following theorem concerning the stability of the dimension of a Reed-Muller code when punctured.

[1] This may happen when for instance the number of errors is greater than t

Theorem 3.4 ([MS86] Chapter 13)**.** *Let j be an integer in range $\{1, \cdots, n\}$ where $n = 2^m$ is the length of the Reed-Muller code $\mathcal{RM}(r, m)$. We have:*

$$\dim\left(\mathcal{RM}(r,m)_j\right) = \sum_{i=0}^{r} \binom{m}{i} = \dim\left(\mathcal{RM}(r,m)\right)$$

The next result is really important because it allows among others to distinguish a Reed-Muller code from a random one.

Proposition 3.5. *Let r and m be two integers such that $0 \leqslant r \leqslant m$ and $2r \leqslant m$. We have*

$$\mathcal{RM}(r,m)^2 = \mathcal{RM}(2r,m)$$

Proof. Let $c_1 = \big(f(a_1), \ldots, f(a_n)\big)$ and $c_2 = \big(g(a_1), \ldots, g(a_n)\big)$ be elements of $\mathcal{RM}(r,m)$ with $\deg f \leqslant r$ and $\deg g \leqslant r$. Hence, $c_1 \star c_2$ is the vector of evaluation $\big(fg(a_1), \ldots, fg(a_n)\big)$ which corresponds to polynomial fg. This means $c_1 \star c_2 \in \mathcal{RM}(2r,m)$. Conversely, each monomial $x_1^{u_1} \ldots x_m^{u_m}$ with $u_i \in \{0,1\}$ and $\sum_i u_i \leqslant 2r$ is the product of two polynomials of degree $\leqslant r$. This proves that a basis of $\mathcal{RM}(2r,m)$ is contained in $\mathcal{RM}(r,m)^2$. □

This proposition allows to observe that for $2r \leqslant m$, the dimension of $\mathcal{RM}(r,m)^2$ is $\sum_{i=0}^{2r} \binom{m}{i}$. For a random $(n,k)-$code $\mathscr{C}$, we have with a high probability,

$$\dim\left(\mathscr{C}^2\right) = \min\left\{n, \binom{k+1}{2}\right\}$$

See [GOT12b, CGG+14] for more details. So one can distinguish a Reed-Muller code from a random one by computing the dimension of the square code. It is supposed that $2r \leqslant m$ but a distinguisher can also be deduce in other cases, using Theorem 2.10. In fact, when $2r > m$ we have

$$(\mathcal{RM}(r,m)^{\perp})^2 = \mathcal{RM}(m-r-1,m)^2 = \mathcal{RM}(2m-2r-2,m)$$

Since $2r > m$ we get $2m - 2r - 2 < m$. That means

$$\dim(\mathcal{RM}(r,m)^{\perp})^2 = \sum_{i=0}^{2m-2r-2} \binom{m}{i}$$

Remark 3.1. By combining Theorem 3.4 and Proposition 3.5, one can easily remark that for an integer j in range $\{1, \cdots, n\}$ we have

$$\dim\left(\mathcal{RM}(r,m)_j\right)^2 = \dim\left(\mathcal{RM}(r,m)^2\right) \tag{3.3}$$

Now we can state the following proposition which is the key result for the sequel of the chapter.

Proposition 3.6. *Let $\boldsymbol{G}$ be a $k \times (n+\ell)$ matrix obtained by inserting ℓ random columns in the generating matrix of a Reed-Muller code $\mathcal{RM}(r,m)$ and let $\mathscr{C}$ be the code spanned by the rows of $\boldsymbol{G}$. Assume that $\ell \leqslant \binom{k+1}{2}$ and $\sum_{i=0}^{2r}\binom{m}{i} \leqslant n$. Then we have:*

$$\sum_{i=0}^{2r}\binom{m}{i} \leqslant \dim \mathscr{C}^2 \leqslant \sum_{i=0}^{2r}\binom{m}{i} + \ell \tag{3.4}$$

Proof. Let $\mathscr{D}_1$ be the code with generating matrix G_1 obtained from $\boldsymbol{G}$ by replacing the last ℓ columns by all-zero columns and let $\mathscr{D}_2$ be the code with generating matrix $\boldsymbol{G}_2$ obtained by replacing in $\boldsymbol{G}$ the first n columns by zero columns. Hence $\boldsymbol{G} = \boldsymbol{G}_1 + \boldsymbol{G}_2$ which implies $\mathscr{D}_1 \subseteq \mathscr{C} \subseteq \mathscr{D}_1 + \mathscr{D}_2$. We have $\mathscr{D}_1 \star \mathscr{D}_2 = 0$ and the following inclusion:

$$\mathscr{D}_1{}^2 \subseteq \mathscr{C}^2 \subseteq \mathscr{D}_1{}^2 + \mathscr{D}_2{}^2 + \mathscr{D}_1 \star \mathscr{D}_2.$$

Observe we have $\mathscr{D}_1 \star \mathscr{D}_2 = 0$. By also remarking $\dim \mathscr{D}_1{}^2 = \dim \mathcal{RM}(2r,m)$ and $\dim \mathscr{D}_2{}^2 = \min\left\{\ell, \binom{k+1}{2}\right\} = \ell$, one can conclude (3.4) is proven. □

From the previous proof, it is obvious that the result remains true with any other family of code. In other words, if $\mathscr{C}$ is a $(n+\ell,k)-$code obtained by inserting ℓ random redundancies in a $(n,k)-$code $\mathscr{D}$ then

$$\dim \mathscr{D}^2 \leqslant \dim \mathscr{C}^2 \leqslant \dim \mathscr{D}^2 + \ell \tag{3.5}$$

We have here inequalities but in practice, we have better than that. For Reed-Muller codes we observed experimentally that for all parameters in [GM13], the upper born is reached. The result is also the same when dealing with punctured Reed-Muller Codes. So in the attack we consider that

$$\dim \mathscr{C}^2 = \dim \mathscr{D}^2 + \ell \tag{3.6}$$

3.3.2 Description of the attack

It is easy for an adversary to use Equation 3.6 to identify the random columns by computing the dimension of $\mathscr{C}^2$ where $\mathscr{C}$ is the code generated by the public matrix $\boldsymbol{G}$ as defined in Section 3.2. We recall that $\mathscr{C}$ is permuted version of a Reed-Muller code $\mathcal{RM}(r,m)$ with ℓ random redundancies at ℓ random positions. We assume that $\sum_{i=0}^{2r}\binom{m}{i} \leqslant n_0$ where $n_0 = 2^m$ and $\ell < \binom{k+1}{2}$ where $k = \sum_{i=0}^{r}\binom{m}{i}$. We now denote by $\mathscr{C}_i$ the code generated by the generating matrix $\boldsymbol{G}_i$ obtained by deleting the i-th column of $\boldsymbol{G}$ (that is to say the punctured code of $\mathscr{C}$ at position i). We also denote by $I \subset \{1,\ldots,n\}$ the set of positions that define the random columns in $\boldsymbol{G}$. We have the following result:

Proposition 3.7. *For any i in $\{1,\ldots,n\}$, two cases occur :*

$$\dim \mathscr{C}_i{}^2 = \begin{cases} \dim \mathscr{C}^2 - 1 & \text{if} \quad i \in I, \\ \dim \mathscr{C}^2 & \text{if} \quad i \notin I. \end{cases} \tag{3.7}$$

Proof. If i belongs to I then, $\mathscr{C}_i$ is a Reed-Muller code $\mathcal{RM}(r,m)$ with $\ell-1$ random redundancies and we have

$$\dim \mathscr{C}_i^2 = \dim \mathcal{RM}(r,m)^2 + \ell - 1 = \dim \mathscr{C}^2 - 1$$

Else, $\mathscr{C}_i$ is the punctured Reed-Muller code $\mathcal{RM}(r,m)_i$ with ℓ random redundancies and then

$$\dim \mathscr{C}_i^2 = \dim \mathcal{RM}(r,m)_i^2 + \ell = \dim \mathcal{RM}(r,m)^2 + \ell = \dim \mathscr{C}^2$$

□

This is the way of distinguishing the random positions of the public code assuming that $\sum_{i=0}^{2r} \binom{m}{i} + \ell \leqslant n$. The set I can then be found and once the set is recovered, it is easy to find the secret $\mathcal{RM}(r,m)$ using usual attacks on Reed-Muller code [MS07, CB13].

3.3.3 Complexity of the attack

Proposition 3.8. *Let $\mathscr{A} \subset \mathbb{F}_q^n$ be a code of dimension k. The complexity of the computation of a basis of $\mathscr{A}^2$ is $O(k^2n^2)$ operations in $\mathbb{F}_q$.*

Proof. The computation, consists first in the computation of $\binom{k+1}{2}$ generators of $\mathscr{A}^2$. This computation costs $O(k^2n)$ operations. Then, we have to apply a Gaussian elimination to a $\binom{k+1}{2} \times n$ matrix, which costs $O(k^2n^2)$ operations. This second step is dominant, which yields the result. □

Our attack relies on the computation of the rank of n square codes so the overall complexity for guessing the random columns is $O(n^5)$ operations in the binary field.

Conclusion

We have studied the security of the modified version of the Sidelnikov scheme [Sid94] given in [GM13] and presented a polynomial-time method that finds the random columns inserted in a secret matrix. This cryptanalysis uses the same approach as [GOT12b, CGG+14] which computes the square codes. The resulting complexity is $O(n^5)$ operations in the binary field. The last step that aims to fully break the scheme consists of using the attacks developed in [MS07, CB13]. This shows that the insertion of random columns in the Sidelnikov scheme does not bring any security improvement and thus open again the problem of finding a good masking technique for Reed-Muller codes.

Chapter 4

Rank Metric Cryptography

Introduction

The concept of rank metric cryptography appeared in [GPT91] where the authors propose a public key encryption scheme using codes in a rank metric framework. They adapted McEliece's general idea [McE78] developed for the Hamming metric to the rank metric context. The key tool in the design is to focus on linear codes having a fast rank-metric decoding algorithm like Gabidulin codes. In this section, we introduce important notions about rank-metric codes and Gabidulin codes and we recall the general principle that underlies all the existing rank-metric encryption schemes.

4.1 Aspects of Rank Metric Codes

For any subfield $\mathbb{K} \subseteq \mathbb{F}$ of a field $\mathbb{F}$ and for any positive integers k and n such that $k \leqslant n$, the $\mathbb{K}$-vector space spanned by $\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k$ where each $\boldsymbol{b}_i \in \mathbb{F}^n$ is denoted by $\sum_{i=1}^{k} \mathbb{K}\, \boldsymbol{b}_i$. The group of invertible matrices of size n over $\mathbb{F}$ is denoted by $\mathsf{GL}_n(\mathbb{F})$.

Definition 4.1 (Rank weight)**.** Let $\boldsymbol{A}$ be a matrix from $\mathcal{M}_{m,n}(\mathbb{F})$ where m and n are positive integers. The *rank weight* of $\boldsymbol{A}$ denoted by $|\boldsymbol{A}|$ is the rank of $\boldsymbol{A}$. The *rank distance* between two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$ from $\mathcal{M}_{m,n}(\mathbb{F})$ is defined as $|\boldsymbol{A} - \boldsymbol{B}|$.

It is a well-known fact that the rank distance on $\mathcal{M}_{m,n}(\mathbb{F})$ has the properties of a metric. But in the context of the rank-metric codes, this rank distance is rather defined for vectors $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$. The idea is to consider the field $\mathbb{F}_{q^m}$ as an $\mathbb{F}_q$-vector space and hence any vector $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ as a matrix from $\mathcal{M}_{m,n}(\mathbb{F}_q)$ by decomposing each entry $x_i \in \mathbb{F}_{q^m}$ into an m-tuple of $\mathbb{F}_q^m$ with respect to an arbitrary basis of $\mathbb{F}_{q^m}$. The rank weight of $\boldsymbol{x}$ also denoted by $|\boldsymbol{x}|$ is then its rank[1] viewed as a matrix of $\mathcal{M}_{m,n}(\mathbb{F}_q)$. Hence, it is possible to define a new metric on $\mathbb{F}_{q^m}^n$ that we recall explicitly in the following.

[1]This rank is of course independent of the choice of the basis of $\mathbb{F}_{q^m}$ since the rank of a matrix is invariant when multiplied by an invertible matrix.

Definition 4.2. Let us consider the finite field extension $\mathbb{F}_{q^m}/\mathbb{F}_q$ of degree $m \geqslant 1$. The *rank weight* of a vector $\boldsymbol{x} = (x_1, x_2, ..., x_n)$ in $\mathbb{F}_{q^m}^n$ denoted by $|\boldsymbol{x}|$ is the dimension of the $\mathbb{F}_q$-vector space generated by $\{x_1, \ldots, x_n\}$

$$|\boldsymbol{x}| = \dim \sum_{i=1}^{n} \mathbb{F}_q x_i. \tag{4.1}$$

The $\mathbb{F}_q-$vector space $\sum_{i=1}^{n} \mathbb{F}_q x_i$ is called the *support* of $\boldsymbol{x}$.

Definition 4.3. The column rank over $\mathbb{F}_q$ of a matrix $\boldsymbol{M}$ from $\mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ is also denoted by $|\boldsymbol{M}|$. It represents the dimension of $\sum_i^n \mathbb{F}_q \boldsymbol{M}_i$ where $\boldsymbol{M}_1, \ldots, \boldsymbol{M}_n$ are the columns of $\boldsymbol{M}$.

In practice, computing the rank weight of a given vector can be done through the bijective mapping $\Phi_{\mathscr{B}}$ associated to a $\mathbb{F}_q-$basis $\mathscr{B} = \{b_1, b_2, ..., b_m\}$ of $\mathbb{F}_{q^m}$ and defined as follows:

$$\begin{array}{rccc} \Phi_{\mathscr{B}}: & \mathbb{F}_{q^m} & \longrightarrow & \mathcal{M}_{m,1}(\mathbb{F}_q) \\ & x = \sum_{i=1}^{m} x_i b_i & \longmapsto & \Phi_{\mathscr{B}}(x) \stackrel{\text{def}}{=} \begin{pmatrix} x_1 \\ x_2 \\ . \\ . \\ . \\ x_m \end{pmatrix} \end{array}$$

$\Phi_{\mathscr{B}}$ can then be extended to vectors $\boldsymbol{x} = (x_1, x_2, ..., x_n) \in \mathbb{F}_{q^m}^n$ by

$$\Phi_{\mathscr{B}}(\boldsymbol{x}) \stackrel{\text{def}}{=} (\Phi_{\mathscr{B}}(x_1), \cdots, \Phi_{\mathscr{B}}(x_n)) \in \mathcal{M}_{m,n}(\mathbb{F}_q).$$

And for a matrix $\boldsymbol{M} = (m_{ij}) \in \mathcal{M}_{k,\ell}(\mathbb{F}_{q^m})$

$$\Phi_{\mathscr{B}}(\boldsymbol{M}) \stackrel{\text{def}}{=} (\Phi_{\mathscr{B}}(m_{ij})) \in \mathcal{M}_{km,\ell}(\mathbb{F}_q)$$

We then have $|\boldsymbol{x}| = \texttt{rank}(\Phi_{\mathscr{B}}(\boldsymbol{x}))$ and $|\boldsymbol{M}| = \texttt{rank}(\Phi_{\mathscr{B}}(\boldsymbol{M}))$

Example 4.1. Let $\mathbb{F}_{2^5} = \mathbb{F}_2 < w >$, $\boldsymbol{x} = (w, w, w, w, w)$, $\boldsymbol{y} = (1, w, w^2, 1 + w^3, w^4)$. We consider $\mathscr{B} = \{1, w, w^2, w^3, w^4\}$ as an $\mathbb{F}_2-$basis of $\mathbb{F}_{2^5}$.

$$|\boldsymbol{x}| = \texttt{rank}(\Phi_{\mathscr{B}}(w), \Phi_{\mathscr{B}}(w), \Phi_{\mathscr{B}}(w), \Phi_{\mathscr{B}}(w), \Phi_{\mathscr{B}}(w)) = \texttt{rank}\begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} = 1.$$

With the same method,

$$|\boldsymbol{y}| = \texttt{rank}\big(\Phi_{\mathscr{B}}(1), \Phi_{\mathscr{B}}(w), \Phi_{\mathscr{B}}(w^2), \Phi_{\mathscr{B}}(1 + w^3), \Phi_{\mathscr{B}}(w^4)\big) = \texttt{rank}\begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix} = 5.$$

If we consider the matrix $\boldsymbol{M}$ given by:

$$\boldsymbol{M} = \begin{pmatrix} 1 & w & w & w & 1+w \\ 1 & w & w^2 & 1+w^3 & 1+w^2 \end{pmatrix}$$

$|\boldsymbol{M}|$ is given by the rank of the matrix

$$\Phi_{\mathscr{B}}(\boldsymbol{M}) = \begin{pmatrix} 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

That is to say $|\boldsymbol{M}| = 4$. One can also remark that the last column of $\boldsymbol{M}$ is a $\mathbb{F}_2-$linear combination of the first column and the third one. Furthermore, the first four columns of $\boldsymbol{M}$ are $\mathbb{F}_2-$linearly independent. Thus $|\boldsymbol{M}| = 4$.

From the above definition we can deduce the following proposition.

Proposition 4.1. *Let $\boldsymbol{M}$ be a matrix from $\mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ and set $s = |\boldsymbol{M}|$ with $s < n$. There exist then $\boldsymbol{M}^*$ in $\mathcal{M}_{k,s}(\mathbb{F}_{q^m})$ with $|\boldsymbol{M}^*| = s$ and $\boldsymbol{T}$ in $\mathsf{GL}_n(\mathbb{F}_q)$ such that:*

$$\boldsymbol{MT} = (\boldsymbol{M}^* \mid \boldsymbol{0}) \tag{4.2}$$

In particular for any $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ such that $|\boldsymbol{x}| = s$ there exists $\boldsymbol{T}$ in $\mathsf{GL}_n(\mathbb{F}_q)$ for which $\boldsymbol{xT} = (\boldsymbol{x}^* \mid \boldsymbol{0})$ where $\boldsymbol{x}^* \in \mathbb{F}_{q^m}^s$ and $|\boldsymbol{x}^*| = s$.

This permits to state the following corollary.

Corollary 4.2. *For any $\boldsymbol{M} \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ and for any $\boldsymbol{m} \in \mathbb{F}_{q^m}^k$*

$$|\boldsymbol{mM}| \leqslant |\boldsymbol{M}| \tag{4.3}$$

Proof. Suppose that $|\boldsymbol{M}| = s$ and let $\boldsymbol{T}$ in $\mathsf{GL}_n(\mathbb{F}_q)$ such that $\boldsymbol{MT} = (\boldsymbol{M}^* \mid \boldsymbol{0})$ with $\boldsymbol{M}^*$ in $\mathcal{M}_{k,s}(\mathbb{F}_{q^m})$. We then have $|\boldsymbol{mM}| = |\boldsymbol{mMT}| = |\boldsymbol{m}(\boldsymbol{M}^* \mid \boldsymbol{0})| \leqslant |\boldsymbol{M}^*| \leqslant s$. □

Rank metric codes

In the sequel, a code $\mathscr{C}$ will be called a rank metric code if the distance used is the rank distance. As in Hamming metric, an important parameter for a rank metric code $\mathscr{C}$ is its minimum distance.

Definition 4.4 (Minimum rank distance)**.** The minimum rank distance d of a rank metric code $\mathscr{C}$ is given by:

$$d = \min\{|\boldsymbol{x}| : \boldsymbol{x} \in \mathscr{C}, \boldsymbol{x} \neq \boldsymbol{0}\}$$

Let $\mathscr{C}$ be a rank metric code of length n and dimension k defined on $\mathbb{F}_{q^m}$. $\boldsymbol{H} \in \mathcal{M}_{n-k,n}(\mathbb{F}_{q^m})$ is a parity check matrix of $\mathscr{C}$. We have the following characterization:

Theorem 4.3. *$\mathscr{C}$ has minimum rank distance d if and only if both of the following conditions are satisfied:*

1. *For any matrix $\boldsymbol{Y}$ belonging to $\mathcal{M}_{d-1,n}(\mathbb{F}_q)$ such that $\mathtt{rank}(\boldsymbol{Y}) = d-1$, we have*
$$\mathtt{rank}(\boldsymbol{Y}\boldsymbol{H}^t) = d-1$$
2. *There exists $\boldsymbol{Y}_0$ belonging to $\mathcal{M}_{d,n}(\mathbb{F}_q)$ with $\mathtt{rank}(\boldsymbol{Y}_0) = d$ and for which*
$$\mathtt{rank}(\boldsymbol{Y}_0\boldsymbol{H}^t) < d$$

Proof. Let us suppose that the minimum distance of $\mathscr{C}$ is d and let $\boldsymbol{c} \in \mathscr{C}$ such that $|\boldsymbol{c}| = d$. Then one can write $\boldsymbol{c} = \boldsymbol{b}\boldsymbol{Y}_0$ with $\boldsymbol{Y}_0 \in \mathcal{M}_{d,n}(\mathbb{F}_q)$ and $\boldsymbol{b} \in \mathbb{F}_{q^m}^d$. Since $\boldsymbol{c}\boldsymbol{H}^t = \boldsymbol{0}$, we have $\boldsymbol{b}\boldsymbol{Y}_0\boldsymbol{H}^t = \boldsymbol{0}$ and since $\boldsymbol{b} \neq \boldsymbol{0}$, this implies that $\mathtt{rank}(\boldsymbol{Y}_0\boldsymbol{H}^t) < d$. So the second point is verified. For the first point, we can use the fact that the code does not contain a non zero word with rank norm less than d. So for any $\boldsymbol{Y} \in \mathcal{M}_{d-1,n}(\mathbb{F}_q)$ with rank $d-1$, the equation

$$(z_1, ..., z_{d-1})\,\boldsymbol{Y}\boldsymbol{H}^t = \boldsymbol{0}$$

has only a trivial solution. That is to say $\mathtt{rank}(\boldsymbol{Y}\boldsymbol{H}^t) = d-1$, so the first point is satisfied. It is obvious that if the first and second conditions are satisfied then the minimum rank distance of $\mathscr{C}$ is d. □

Maximum Rank Distance Codes

Maximum Rank Distance (MRD) codes in rank metric are the equivalents of Maximum Distance Separable (MDS) codes in hamming metric. In this sub-section, we define and give some characterizations of MRD codes. We start with the following proposition.

Proposition 4.4. *Let $\mathscr{C}$ be an $(n,k)-$code on $\mathbb{F}_{q^m}$ and d its minimum rank distance. Assuming $n \leqslant m$ we have:*

$$d \leqslant n - k + 1$$

Proof. Let $\boldsymbol{x}$ be a non-zero codeword of $\mathscr{C}$. It is obvious that $|\boldsymbol{x}| \leqslant w(\boldsymbol{x})$. If we denote by d_H the minimum hamming distance of $\mathscr{C}$, then we have $d \leqslant d_H$ and from the Singleton bound we get $d \leqslant d_H \leqslant n - k + 1$. □

We will assume in the sequel that any $(n,k)-$ rank-metric code on $\mathbb{F}_{q^m}$ satisfies $n \leqslant m$.

Definition 4.5. A $(n,k,d)-$code $\mathscr{C}$ is a Maximum Rank Distance code if it minimum rank distance satisfies

$$d = n - k + 1$$

The following proposition gives a characterization for MRD codes.

Proposition 4.5. *Let $\mathscr{C}$ be an $(n,k)-$code with parity check matrix $\boldsymbol{H}$. $\mathscr{C}$ is MRD if and only if for any $\boldsymbol{Y} \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$ of rank $n-k$,*

$$\mathtt{rank}(\boldsymbol{Y}\boldsymbol{H}^t) = n - k$$

Proof. If for any $\boldsymbol{Y} \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$ with rank $n-k$ we have $\mathtt{rank}(\boldsymbol{Y}\boldsymbol{H}^t) = n-k$ then, from Theorem 4.3, $d \geqslant n-k+1$ and thus $d = n-k+1$. Conversely, if $d = n-k+1$ then $d-1 = n-k$ and we get the result from Theorem 4.3. □

Theorem 4.6. *Let $\mathscr{C}$ be a $(n,k)-$code. $\mathscr{C}$ is an MRD code if and only if $\mathscr{C}^\perp$ is an MRD code.*

Proof. Let $\mathscr{C}$ be an MRD code and $\boldsymbol{H}$ a generator matrix of $\mathscr{C}^\perp$. From proposition 4.5, it follows that for any $\boldsymbol{Y} \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$ of rank $n-k$, $\mathtt{rank}(\boldsymbol{Y}\boldsymbol{H}^t) = n-k$. This implies that for any non zero codeword $\boldsymbol{h} \in \mathscr{C}^\perp$ and for any $\boldsymbol{Y} \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$ of rank $n-k$, $\boldsymbol{Y}\boldsymbol{h}^t \neq \boldsymbol{0}$. Assume that there exists $\boldsymbol{h} \in \mathscr{C}^\perp$ with $\mathtt{rank}(\boldsymbol{h}) \leqslant k$. Then

$$\boldsymbol{h} = \boldsymbol{b}\boldsymbol{X} = (b_1, ..., b_k)\,\boldsymbol{X} \text{ with } \boldsymbol{X} \in \mathcal{M}_{k,n}(\mathbb{F}_q) \text{ and } \mathtt{rank}(\boldsymbol{X}) = k.$$

So for any $\boldsymbol{Y}$ in $\mathcal{M}_{n-k,n}(\mathbb{F}_q)$,

$$\boldsymbol{Y}\boldsymbol{X}^t\boldsymbol{b}^t \neq \boldsymbol{0}. \tag{4.4}$$

Furthermore, for any $\boldsymbol{X}$ in $\mathcal{M}_{k,n}(\mathbb{F}_q)$ of rank k, there exists an orthogonal matrix $\boldsymbol{Y}_0 \in \mathcal{M}_{n-k,n}(\mathbb{F}_q)$ such that $\boldsymbol{Y}_0\boldsymbol{X}^t = \boldsymbol{0}$. This implies that $\boldsymbol{Y}_0\boldsymbol{X}^t\boldsymbol{b}^t = \boldsymbol{0}$ and that is a contradiction with (4.4). So, $\mathscr{C}^\perp$ does not contain a non-zero codeword of rank less than (or equal to) k. We then deduce that the minimum distance of $\mathscr{C}^\perp$ is $k+1$ and thus, $\mathscr{C}^\perp$ is an MRD code. To finish, let's suppose that $\mathscr{C}^\perp$ is an MRD code, then $\mathscr{C}^{\perp\perp}$ is also an MRD code and since $\mathscr{C}^{\perp\perp} = \mathscr{C}$, this implies that $\mathscr{C}$ is an MRD code. □

From this theorem we can deduce another characterization of MRD codes, using generating matrices.

Corollary 4.7. *Let $\mathscr{C}$ be a $(n,k)-$code and $\boldsymbol{G}$ a generating matrix of $\mathscr{C}$. $\mathscr{C}$ is MRD if and only if for any $\boldsymbol{X} \in \mathcal{M}_{k,n}(\mathbb{F}_q)$ of rank k,*

$$\mathtt{rank}(\boldsymbol{X}\boldsymbol{G}^t) = k$$

4.1.1 Hardness of the Rank Decoding Problem

The rank decoding problem is the equivalent of the hamming decoding problem in the rank metric context. The search version can be defined as follows:

Definition 4.6 (Rank Decoding Problem $\mathcal{RD}$)**.** Let $\boldsymbol{G}$ be a full-rank matrix belonging to $\mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ with $k \leqslant n$, $\boldsymbol{y}$ an element of $\mathbb{F}_{q^m}^n$ and t an integer. The *Rank Decoding Problem* is to find $\boldsymbol{e}$ in $\mathbb{F}_{q^m}^n$ and $\boldsymbol{m}$ in $\mathbb{F}_{q^m}^k$ such that $\boldsymbol{y} = \boldsymbol{mG} + \boldsymbol{e}$ with $|\boldsymbol{e}| \leqslant t$. This problem will be denoted as the $\mathcal{RD}_{q,m,n,k,t}$ problem.

The decisional version of this problem is the following:

Definition 4.7 (Decisional Rank Decoding Problem)**.** Let $\boldsymbol{G}$ be a full-rank matrix belonging to $\mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ with $k \leqslant n$ and t an integer. Consider $\boldsymbol{y} \in \mathbb{F}_{q^m}^n$ and $\mathbf{D}_1$ be the following distribution:

$$\mathbf{D}_1 = \{\boldsymbol{xG} + \boldsymbol{e}, \boldsymbol{x} \in \mathbb{F}_{q^m}^k, \boldsymbol{e} \in \mathbb{F}_{q^m}^n \text{ with } |\boldsymbol{e}| \leqslant t\}$$

The *Decisional Rank Decoding* Problem $\mathcal{DRD}_{q,m,n,k,t}$ is to distinguish whether $\boldsymbol{y}$ belongs to $\mathbf{D}_1$ or not.

The dual variant of this problem is called the Decisional Rank Syndrome Decoding Problem and is equivalent to the $\mathcal{DRD}$ problem :

Definition 4.8 (Decisional Rank Syndrome Decoding Problem $\mathcal{DRSD}$)**.** Let $\boldsymbol{H} \in \mathcal{M}_{n-k,n}(\mathbb{F}_{q^m})$ where n and k are positive integers with $k \leqslant n$, t another integer. Let $\boldsymbol{s} \in \mathbb{F}_{q^m}^{n-k}$ and $\mathbf{D}_2$ be the following distribution:

$$\mathbf{D}_2 = \{\boldsymbol{He}^t, \boldsymbol{e} \in \mathbb{F}_{q^m}^n \text{ with } |\boldsymbol{e}| \leqslant t\}$$

The *Decisional Rank Syndrome Decoding* Problem $\mathcal{DRSD}_{q,m,n,k,t}$ is to distinguish whether $\boldsymbol{s}$ belongs to $\mathbf{D}_2$ or not.

These problems was recently proven to be NP-hard [GZ16]. In the following paragraphs, we give some references about the best algorithms for solving the rank decoding problem.

4.1.2 Algorithms for Solving the Rank Decoding Problem

Existing algorithms that solve the $\mathcal{RD}_{q,m,n,k,t}$ problem can be divided in two classes: Combinatorial algorithms and algebraic algorithms.

Combinatorial Algorithms

These algorithms consider the properties of rank metric on a combinatorial point of view in order to recover the support or the coordinates of the error vector. The problem being to find a word $\boldsymbol{e}$ from $\mathbb{F}_{q^m}^n$ with $|\boldsymbol{e}| \leqslant t$ such that $\boldsymbol{He}^t = \boldsymbol{s}^t$, one can remark that the coordinates of $\boldsymbol{e}$ are elements of an $\mathbb{F}_q-$vector subspace $\mathscr{V}$ of $\mathbb{F}_q^m$ with $\dim \mathscr{V} = t$ ($\mathscr{V}$ is the support of $\boldsymbol{e}$). Let $\boldsymbol{b} = (b_1, \cdots, b_t)$ be a basis of $\mathscr{V}$ considered as a vector of length t. Thus $\boldsymbol{e}$ can be reformulated as $\boldsymbol{e} = \boldsymbol{bE}$ where $\boldsymbol{E}$ belongs to $\mathcal{M}_{t,n}(\mathbb{F}_q)$. Let $\Omega = (\omega_1, \cdots, \omega_m)$ be a $\mathbb{F}_q-$basis of $\mathbb{F}_{q^m}$. $\boldsymbol{b}$ can be also represented as a $m \times t$ matrix using de mapping Φ_Ω.

Chabaud-Stern algorithm The first idea from Chabaud-Stern [CS96] is to enumerate all the different possible basis $\boldsymbol{b}$ of t vectors and for each $\boldsymbol{b}$, solve the system $\boldsymbol{H}\boldsymbol{E}^t\boldsymbol{b}^t = \boldsymbol{s}^t$ in which the unknowns are the entries of $\boldsymbol{E}$. One can remark that the system is linear once $\boldsymbol{b}$ is known. The complexity of the enumeration phase is q^{mt} operations by testing all the representations of each basis and can be reduced to $q^{(m-t)(t-1)}$ operations by testing only one representation of each basis to enumerate. This gives a global complexity of $(nt+m)^3\, q^{(m-t)(t-1)}$ operations in $\mathbb{F}_q$.

Ourivski and Johannsson algorithms The approach of [OJ02] consists to consider the equation $\boldsymbol{y} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{e}$ instead of $\boldsymbol{H}\boldsymbol{e}^t = \boldsymbol{s}$, where $\boldsymbol{y}$ is the received word, $\boldsymbol{m} \in \mathbb{F}_{q^m}^k$ and $\boldsymbol{e}$ an error of rank t. It follows that

$$\begin{pmatrix} \boldsymbol{G} \\ \boldsymbol{y} \end{pmatrix} = \begin{pmatrix} \boldsymbol{I}_k & 0 \\ \boldsymbol{x} & 1 \end{pmatrix} \begin{pmatrix} \boldsymbol{G} \\ \boldsymbol{e} \end{pmatrix} \tag{4.5}$$

Let $\mathscr{C}_{\boldsymbol{e}}$ be the code generated by $\begin{pmatrix} \boldsymbol{G} \\ \boldsymbol{y} \end{pmatrix}$. From Eq.4.5, we thus deduce the following inclusion:

$$\{\lambda \boldsymbol{e} \ : \ \lambda \in \mathbb{F}_{q^m}^*\} \subset \mathscr{C}_{\boldsymbol{e}} \tag{4.6}$$

The idea is then to find an element $\boldsymbol{e}' = \lambda\boldsymbol{e} \in \mathscr{C}_{\boldsymbol{e}}$ of weight t and deduce λ by computing the syndromes $\boldsymbol{e}'\boldsymbol{H}^t$ and $\boldsymbol{y}\boldsymbol{H}^t = \boldsymbol{e}\boldsymbol{H}^t$. Let $\boldsymbol{G}_e = (\boldsymbol{I}_{k+1} \mid \boldsymbol{R})$ be the systematic generator matrix of $\mathscr{C}_{\boldsymbol{e}}$. The fact that $\boldsymbol{e}$ belongs to $\mathscr{C}_{\boldsymbol{e}}$ implies that $\boldsymbol{e} = \boldsymbol{e}_1\boldsymbol{G}_e = (\boldsymbol{e}_1 \mid \boldsymbol{e}_1\boldsymbol{R})$, $\boldsymbol{e}_1$ being the vector composed by the first $k+1$ coordinates of $\boldsymbol{e}$ and hence satisfies $|\boldsymbol{e}_1| \leqslant t$. Furthermore, the fact that $|\boldsymbol{e}| = t$ implies that there exist an incomplete basis $\{b_1, \cdots, b_t\}$ of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ and a full rank matrix $\boldsymbol{A} \in \mathcal{M}_{t,n}(\mathbb{F}_q)$ such that $\boldsymbol{e} = (b_1, \cdots, b_t)\,\boldsymbol{A}$. Assume that $\boldsymbol{A} = (\boldsymbol{A}_1 \mid \boldsymbol{A}_2)$ with $\boldsymbol{A}_1 \in \mathcal{M}_{t,k+1}(\mathbb{F}_q)$. We have:

$$\boldsymbol{e} = (\boldsymbol{e}_1 \mid \boldsymbol{e}_1\boldsymbol{R}) = (b_1, \cdots, b_t)\,(\boldsymbol{A}_1 \mid \boldsymbol{A}_2) \tag{4.7}$$

Eq. 4.7 implies that $\boldsymbol{e}_1 = (b_1, \cdots, b_t)\,\boldsymbol{A}_1$ and $\boldsymbol{e}_1\boldsymbol{R} = (b_1, \cdots, b_t)\,\boldsymbol{A}_2$. Combining both equations allows to get

$$(b_1, \cdots, b_t)\,\boldsymbol{A}_2 = (b_1, \cdots, b_t)\,\boldsymbol{A}_1\boldsymbol{R} \tag{4.8}$$

Since it is enough to get $\lambda\boldsymbol{e}$ for any $\lambda \in \mathbb{F}_{q^m}^*$, we can choose $b_1 = 1$. Eq. 4.8 is then a system of $n-k-1$ equations with $nt+t-1$ unknowns, namely the components a_{ij} of the matrix $\boldsymbol{A}$ and the remaining $b_i \in \mathbb{F}_{q^m}$, $i = 2, ..., t$. Let $\Omega = \{\omega_1, \cdots, \omega_m\}$ be an $\mathbb{F}_q-$ basis of $\mathbb{F}_{q^m}$. By expressing each coordinate of $\boldsymbol{b}$ and $\boldsymbol{R}$ in the basis Ω, Eq. 4.8 can be rewritten as a quadratic system of $m\,(n-k-1)$ equations in $nt + m(t-1)$ unknowns over $\mathbb{F}_q$. Ourivski and Johannsson [OJ02] proposed two strategies for solving this system. One with $(k+t)^3\, t^3 q^{(m-t)(t-1)+2}$ operations in $\mathbb{F}_q$ and the second with $(tm)^3\, q^{(t-1)(k+1)+2}$ operations on $\mathbb{F}_q$. The general technique consists to guess the values of some unknowns contributing to quadratic terms and solve the resulting linear system.

Gaborit-Ruatta-Shreck algorithm In [GRS16], a new combinatorial algorithm is presented. This algorithm can be seen as an adaptation to the rank metric of the information set decoding, but in its dual version, namely the *error support attack*. The idea is to guess a subspace $\mathscr{V}'$ containing the error support $\mathscr{V}$ and find $\boldsymbol{e}$ by solving a linear system derives from the syndrome equations. An important point of this algorithm is the introduction of the ratio m/n in the exponent of the complexity. The new complexity is $\min\{O((n-k)^3m^3q^{(t-1)\lfloor(k+1)m/n\rfloor}, O((n-k)^3m^3q^{t\lfloor km/n\rfloor}\}$ operations on $\mathbb{F}_q$. This gives a major difference in the cases $n > m$.

Algebraic Algorithms

The main idea for algebraic algorithms is to translate the notion of rank into an algebraic setting. The first approach from [LdVP06] considers Eq. 4.8 together with the syndrome equations

$$\begin{cases} (b_1, ..., b_t)\,\boldsymbol{A}\boldsymbol{H}^t &= \boldsymbol{y}\boldsymbol{H}^t \\ (b_1, ..., b_t)\,\boldsymbol{A}_2 &= (b_1, \cdots, b_t)\,\boldsymbol{A}_1\boldsymbol{R} \end{cases} \tag{4.9}$$

Writing Eq. 4.9 in the basis Ω gives a system of $m(2(n-k)-1)$ equations in $nt+m(r-1)$ unknowns over $\mathbb{F}_q$ and is solved with Gröbner basis techniques. A new setting based on linearized polynomials was recently proposed in [GRS16] and can allow to solve the problem in $O(((t+1)(k+1)-1)^3)$ operations in $\mathbb{F}_{q^m}$ with linearization technique when the condition $n \geqslant (t+1)(k+1)-1$ is satisfied. In general, it is shown in [GRS16] that the $\mathcal{RD}$ problem can be solved by an hybrid approach (algebraic and combinatorial) with at most $t^3k^3q^{tk}$ operations in $\mathbb{F}_q$ assuming that $\lceil\frac{(t+1)(k+1)-(n+1)}{t}\rceil \leqslant k$. Another type of algebraic modelling which can be solved with Gröbner basis techniques can be found in [FLdP08]. But since the attack consider algebraic systems on the base field $\mathbb{F}_q$, the number of unknowns is quadratic in the length of the code. The global complexity of Gröbner basis attacks being exponential in the number of unknowns, it implies that the complexity is in general exponential when dealing with cryptographic parameters.

In the following, we present a special family of MRD codes known as Gabidulin codes.

4.1.3 Gabidulin Codes

In this section and in the sequel, for any x in $\mathbb{F}_{q^m}$ and for any integer i, the quantity x^{q^i} is denoted by $x^{[i]}$. This notation is extended to vectors $\boldsymbol{x}^{[i]} = (x_1^{[i]}, \ldots, x_n^{[i]})$ and matrices $\boldsymbol{M}^{[i]} = \left(m_{ij}^{[i]}\right)$. The following lemma will be useful in the sequel.

Lemma 4.8. *For any $\boldsymbol{A} \in \mathcal{M}_{\ell,s}(\mathbb{F}_{q^m})$ and $\boldsymbol{B} \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$, and for any α and β in $\mathbb{F}_q$:*

1. *If $\ell = k$ and $s = n$ then*

$$(\alpha\boldsymbol{A} + \beta\boldsymbol{B})^{[i]} = \alpha\boldsymbol{A}^{[i]} + \beta\boldsymbol{B}^{[i]}$$

2. If $s = k$ then

$$(\boldsymbol{AB})^{[i]} = \boldsymbol{A}^{[i]}\boldsymbol{B}^{[i]}.$$

In particular if $\boldsymbol{S}$ is in $\mathsf{GL}_n(\mathbb{F}_{q^m})$ then $\boldsymbol{S}^{[i]}$ also belongs to $\mathsf{GL}_n(\mathbb{F}_{q^m})$ and

$$(\boldsymbol{S}^{[i]})^{-1} = \left(\boldsymbol{S}^{-1}\right)^{[i]}$$

Proof. The proof of the two points comes directly from the properties of the Frobenius operators (multiplicative and $\mathbb{F}_q-$linear). To finish, remark that for $\boldsymbol{S}$ in $\mathsf{GL}_n(\mathbb{F}_{q^m})$, since $\boldsymbol{S}\boldsymbol{S}^{-1} = \boldsymbol{I}_n$ we also have $\boldsymbol{S}^{[i]}\left(\boldsymbol{S}^{-1}\right)^{[i]} = \boldsymbol{I}_n$. This implies that $\boldsymbol{S}^{[i]}$ belongs to $\mathsf{GL}_n(\mathbb{F}_{q^m})$ and $(\boldsymbol{S}^{[i]})^{-1} = \left(\boldsymbol{S}^{-1}\right)^{[i]}$ □

We introduce now an important family of codes known for having an efficient decoding algorithm.

Definition 4.9 (Gabidulin code)**.** Let $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ such that $|\boldsymbol{g}| = n$. The $(n,k)-$Gabidulin code denoted by $\mathscr{G}_k(\boldsymbol{g})$ is the code with a generator matrix $\boldsymbol{G}$ where

$$\boldsymbol{G} = \begin{pmatrix} g_1^{[0]} & \cdots & g_n^{[0]} \\ \vdots & & \vdots \\ g_1^{[k-1]} & \cdots & g_n^{[k-1]} \end{pmatrix}. \tag{4.10}$$

A matrix of the form (4.10) is called a $q-$Vandermonde matrix.

Lemma 4.9. *Gabidulin codes are Maximum Rank Distance (MRD) codes.*

Proof. It is sufficient to establish that for any $\boldsymbol{X} \in \mathcal{M}_{k,n}(\mathbb{F}_q)$ of rank k, $\mathtt{rank}(\boldsymbol{G}\boldsymbol{X}^t)$ is also equal to k. For any $\boldsymbol{X}$ in $\mathcal{M}_{k,n}(\mathbb{F}_q)$, the matrix $\boldsymbol{G}\boldsymbol{X}^t$ is a square matrix of the form

$$\boldsymbol{G}\boldsymbol{X}^t = \begin{pmatrix} f_1^{[0]} & \cdots & f_k^{[0]} \\ \vdots & & \vdots \\ f_1^{[k-1]} & \cdots & f_k^{[k-1]} \end{pmatrix}.$$

with $(f_1, \cdots, f_k) = (g_1, \cdots, g_n)\,\boldsymbol{X}^t$. Since $|\boldsymbol{g}| = n$ then $|\boldsymbol{f}| = \min\{n, \mathtt{rank}(\boldsymbol{X})\} = k$ and we deduce that $\mathtt{rank}(\boldsymbol{G}\boldsymbol{X}^t) = k$. □

From this lemma we can deduce that the error correction capability of a Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$ is $\lfloor \frac{1}{2}(n-k) \rfloor$. It can also be used to prove the following proposition:

Proposition 4.10. *The dual of $\mathscr{G}_k(\boldsymbol{g})$ is the Gabidulin code $\mathscr{G}_{n-k}(\boldsymbol{h})$ where $\boldsymbol{h} = \boldsymbol{y}^{[-(n-k-1)]}$ and $\boldsymbol{y}$ belongs to $\mathscr{G}_{n-1}(\boldsymbol{g})^{\perp}$.*

Proof. One can remark from (4.10) that

$$\mathscr{G}_{n-1}(\boldsymbol{g})^{\perp} \subset \mathscr{G}_{n-2}(\boldsymbol{g})^{\perp} \subset \cdots \subset \mathscr{G}_k(\boldsymbol{g})^{\perp} \tag{4.11}$$

Since $\mathscr{G}_{n-1}(\boldsymbol{g})^{\perp}$ is an MRD code of dimension 1, it minimum distance is $d = n$. Thus any non zero element of $\mathscr{G}_{n-1}(\boldsymbol{g})^{\perp}$ is of rank weight n. Let $\boldsymbol{y}$ be a non zero element of $\mathscr{G}_{n-1}(\boldsymbol{g})^{\perp}$. We have for all $i \in \{0, \ldots, n-2\}$

$$\sum_{j=1}^{n} y_j g_j^{[i]} = 0.$$

This implies that,

$$\forall i \in \{0, \ldots, n-2\}, \quad \sum_{j=1}^{n} y_j^{[-1]} g_j^{[i-1]} = 0$$

In particular,

$$\forall i \in \{0, \ldots, n-3\}, \quad \sum_{j=1}^{n} y_j^{[-1]} g_j^{[i]} = 0$$

Thus, $\boldsymbol{y}^{[-1]}$ belongs to $\mathscr{G}_{n-2}(\boldsymbol{g})^{\perp}$. We can deduce by induction that for all u in $\{0, \ldots, n-1\}$,

$$\boldsymbol{y}^{[-u]} \in \mathscr{G}_{n-1-u}(\boldsymbol{g})^{\perp}$$

and for a given u in $\{0, \ldots, n-1\}$ we have

$$\forall i \in \{0, \ldots, u\} \quad \boldsymbol{y}^{[-u+i]} \in \mathscr{G}_{n-1-u+i}(\boldsymbol{g})^{\perp} \subset \mathscr{G}_{n-1-u}(\boldsymbol{g})^{\perp}.$$

For $u = n-k-1$ and $\boldsymbol{h} = \boldsymbol{y}^{[-u]}$ we have $\boldsymbol{h}^{[i]} \in \mathscr{G}_k(\boldsymbol{g})^{\perp}$ for all i in $\{0, \ldots, n-k-1\}$. That is to say

$$\mathscr{G}_k(\boldsymbol{g})^{\perp} = \mathscr{G}_{n-k}(\boldsymbol{h})$$

□

Gabidulin codes are known to possess a fast decoding algorithm that can decode errors of weight t provided that $t \leqslant \lfloor \frac{1}{2}(n-k) \rfloor$. We end this section by an important well-known property about Gabidulin codes.

Proposition 4.11. *Let $\mathscr{G}_k(\boldsymbol{g})$ be a Gabidulin code of length n with generator matrix $\boldsymbol{G}$ and $\boldsymbol{T} \in \mathsf{GL}_n(\mathbb{F}_q)$. Then $\boldsymbol{GT}$ is a generator matrix of the Gabidulin code $\mathscr{G}_k(\boldsymbol{gT})$*

Proof. From Lemma 4.8, we have $(\boldsymbol{gT})^{[i]} = \boldsymbol{g}^{[i]}\boldsymbol{T}$. □

4.2 Rank Metric Encryption Schemes

In this section, we recall the general principle that underlies all the existing rank encryption metric schemes based on Gabidulin codes. During the key generation phase, the integers k, ℓ, n and m are chosen such that $k < n \leqslant m$ and $0 \leqslant \ell \ll n$. It then randomly picks $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ with $|\boldsymbol{g}| = n$ and defines $\boldsymbol{G} \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ as in (4.10), that is to say $\boldsymbol{G}$ is a generator matrix of the Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$. The

error-correcting capacity of $\mathscr{G}_k(\boldsymbol{g})$ is denoted by $t \stackrel{\text{def}}{=} \lfloor \frac{1}{2}(n-k) \rfloor$. An important step in the key generation is the "hiding" phase where $\boldsymbol{G}$ undergoes a transformation to mask the algebraic structure of Gabidulin codes. This transformation is actually a probabilistic algorithm that adds some randomness to its input. Originally, the authors in [GPT91] proposed to use a *distortion* transformation

$$\mathcal{D} : \mathbb{F}_{q^m}^{k \times n} \longrightarrow \mathbb{F}_{q^m}^{k \times n}$$

$\mathcal{D}$ sends any $\boldsymbol{G}$ to

$$\mathcal{D}(\boldsymbol{G}) = \boldsymbol{S}(\boldsymbol{G} + \boldsymbol{X})$$

where $\boldsymbol{X}$ is a random matrix from $\mathbb{F}_{q^m}^{k \times n}$ with a prescribed rank $t_{\boldsymbol{X}}$ and $\boldsymbol{S}$ is an invertible matrix. The public key is then $\boldsymbol{G}_{\text{pub}} = \mathcal{D}(\boldsymbol{G})$ with the parameter

$$t_{\text{pub}} = t - t_{\boldsymbol{X}}$$

While the private key is $(\boldsymbol{S}, \boldsymbol{G})$. The encryption algorithm takes as input a plaintext $\boldsymbol{m} \in \mathbb{F}_{q^m}^k$ and generates a random $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ such that $|\boldsymbol{e}| \leqslant t_{\text{pub}}$ in order to compute the ciphertext

$$\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G}_{\text{pub}} + \boldsymbol{e}.$$

In the decryption step the decoding algorithm of the Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$ is applied to the ciphertext. This word can be decoded since the underlying codeword is corrupted by the error vector $\boldsymbol{m}\boldsymbol{S}\boldsymbol{X} + \boldsymbol{e}$ whose rank weight is less than t since by Corollary 4.2 we have

$$|\boldsymbol{m}\boldsymbol{S}\boldsymbol{X}| \leqslant t_{\boldsymbol{X}} \quad \text{and} \quad |\boldsymbol{m}\boldsymbol{S}\boldsymbol{X} + \boldsymbol{e}| \leqslant |\boldsymbol{m}\boldsymbol{S}\boldsymbol{X}| + |\boldsymbol{e}| \leqslant t$$

However, Gibson proved [Gib95, Gib96] that the GPT encryption scheme [GPT91] is vulnerable to a polynomial time key recovery attack. Consequently, Gabidulin and Ourivski proposed in [GO01] a reparation by considering a more general hiding transformation combining a distortion matrix $\boldsymbol{X}$ and a right column scrambler $\boldsymbol{P}$. The hidden generator matrix is more precisely of the form:

$$\mathcal{D}(\boldsymbol{G}) = \boldsymbol{S}\left(\boldsymbol{X}_1 \mid \boldsymbol{G} + \boldsymbol{X}_2\right)\boldsymbol{P} \tag{4.12}$$

where $\boldsymbol{X}_1 \in \mathcal{M}_{k,\ell}(\mathbb{F}_{q^m})$, $\boldsymbol{X}_2 \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ such that $|\boldsymbol{X}_2| < t$ and $\boldsymbol{P} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$. The public generator matrix is again $\boldsymbol{G}_{\text{pub}} \stackrel{\text{def}}{=} \mathcal{D}(\boldsymbol{G})$ which constitutes the public key with the public parameter $t_{\text{pub}} \stackrel{\text{def}}{=} t - t_2$ where $t_2 \stackrel{\text{def}}{=} |\boldsymbol{X}_2|$. The decryption computes $\boldsymbol{P}^{-1} = (\boldsymbol{Q}_1 \mid \boldsymbol{Q}_2)$ where $\boldsymbol{Q}_1 \in \mathcal{M}_{(n+\ell),\ell}(\mathbb{F}_q)$ and $\boldsymbol{Q}_2 \in \mathcal{M}_{(n+\ell),n}(\mathbb{F}_q)$. The last n components of $\boldsymbol{c}\boldsymbol{P}^{-1}$ is the vector $\boldsymbol{m}\boldsymbol{S}\boldsymbol{G} + \boldsymbol{m}\boldsymbol{S}\boldsymbol{X}_2 + \boldsymbol{e}\boldsymbol{Q}_2$ and since $|\boldsymbol{e}\boldsymbol{Q}_2| \leqslant |\boldsymbol{e}|$ and $|\boldsymbol{m}\boldsymbol{S}\boldsymbol{X}_2| \leqslant |\boldsymbol{X}_2|$, it follows that $|\boldsymbol{m}\boldsymbol{S}\boldsymbol{X}_2 + \boldsymbol{e}\boldsymbol{Q}_2| \leqslant t$. Applying a fast decoding algorithm to the last n components of $\boldsymbol{c}\boldsymbol{P}^{-1}$ allows the legitimate user to get $\boldsymbol{m}\boldsymbol{S}$ and easily $\boldsymbol{m}$.

We now state this result about Gabidulin and Ouriviski reparation which proves that we can always consider $\boldsymbol{X}_2 = 0$.

Proposition 4.12. *Let* $\boldsymbol{G}_{\text{pub}}$ *be as in* (4.12) *and assume that* $|\boldsymbol{X}_2| = t_2$. *There exist* $\boldsymbol{P}^* \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$, $\boldsymbol{X}^* \in \mathcal{M}_{k,(\ell+t_2)}(\mathbb{F}_{q^m})$ *and a matrix* $\boldsymbol{G}^*$ *that generates an* $(n-t_2,k)-$*Gabidulin code* $\mathscr{G}_k(\boldsymbol{g}^*)$ *such that*

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}\,(\boldsymbol{X}^* \mid \boldsymbol{G}^*)\,\boldsymbol{P}^*. \tag{4.13}$$

Furthermore, the error correction capability t^* *of* $\mathscr{G}_k(\boldsymbol{g}^*)$ *is equal to* $t - \frac{1}{2}t_2$, *and hence* $t^* > t_{\text{pub}}$.

Proof. Since $|\boldsymbol{X}_2| = t_2$ then by Proposition 4.1 there exist $\boldsymbol{T}_2$ in $\mathsf{GL}_n(\mathbb{F}_q)$ and $\boldsymbol{X}'_2$ in $\mathcal{M}_{k,t_2}(\mathbb{F}_{q^m})$ such that $\boldsymbol{X}_2\boldsymbol{T}_2 = (\boldsymbol{X}'_2 \mid \boldsymbol{0})$. So by letting $\boldsymbol{T} = \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{T}_2 \end{pmatrix}$ we then have:

$$\begin{aligned} \boldsymbol{G}_{\text{pub}} = \boldsymbol{S}\,(\boldsymbol{X}_1 \mid \boldsymbol{G} + \boldsymbol{X}_2)\,\boldsymbol{P} &= \boldsymbol{S}\,(\boldsymbol{X}_1 \mid \boldsymbol{G}\boldsymbol{T}_2 + \boldsymbol{X}_2\boldsymbol{T}_2)\,\boldsymbol{T}^{-1}\boldsymbol{P} \\ &= \boldsymbol{S}\,(\boldsymbol{X}_1 \mid \boldsymbol{G}' + \boldsymbol{X}_2\boldsymbol{T}_2)\,\boldsymbol{Q} \end{aligned}$$

where $\boldsymbol{G}' = \boldsymbol{G}\boldsymbol{T}_2$ and $\boldsymbol{Q} = \boldsymbol{T}^{-1}\boldsymbol{P}$. Note that $\boldsymbol{G}'$ generates the $(n,k)-$Gabidulin code $\mathscr{G}_k(\boldsymbol{g}')$ with $\boldsymbol{g}' = \boldsymbol{g}\boldsymbol{T}_2 = (g'_1,\ldots,g'_n)$. Let us decompose $\boldsymbol{G}'$ as $(\boldsymbol{G}'_1 \mid \boldsymbol{G}'_2)$ where $\boldsymbol{G}'_1 \in \mathcal{M}_{k,t_2}(\mathbb{F}_{q^m})$ and $\boldsymbol{G}'_2 \in \mathcal{M}_{k,(n-t_2)}(\mathbb{F}_{q^m})$ we then have:

$$\boldsymbol{G}' + \boldsymbol{X}_2\boldsymbol{T}_2 = (\boldsymbol{G}'_1 + \boldsymbol{X}'_2 \mid \boldsymbol{G}'_2)$$

By setting $\boldsymbol{X} = (\boldsymbol{X}_1 \mid \boldsymbol{G}'_1 + \boldsymbol{X}'_2)$ we get (4.13) and $\boldsymbol{G}'_2$ generates the $(n-t_2,k)-$Gabidulin $\mathscr{G}_k(\boldsymbol{g}'_2)$ where $\boldsymbol{g}'_2 = (g'_{t_2+1},\ldots,g'_n)$. The error-correction capability t^* of $\mathscr{G}_k(\boldsymbol{g}'_2)$ is given by $t^* = \frac{1}{2}(n-t_2-k) = t - \frac{1}{2}t_2$ which implies $t^* > t - t_2$. □

The first important consequence of Proposition 4.12 is the possibility for a cryptanalysist who is able to derive $(\boldsymbol{S},\boldsymbol{G}^*,\boldsymbol{P}^*)$ from $\boldsymbol{G}_{\text{pub}}$ so that (4.13) is satisfied to decipher any ciphertext $\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G}_{\text{pub}} + \boldsymbol{e}$ with $|\boldsymbol{e}| \leqslant t_{\text{pub}}$. Thus any successful structural attack on the description (4.13) leads to a successful attack on (4.12) and conversely since (4.13) corresponds to the special case where $\boldsymbol{X}_2 = \boldsymbol{0}$. Therefore the security of the scheme given [GO01] is equivalent to the one of a scheme where $\boldsymbol{X}_2 = \boldsymbol{0}$.

4.2.1 Distinguishing Properties of Gabidulin Codes

We recall important algebraic properties about Gabidulin codes. It will explain why many attacks occur when the underlying code is a Gabidulin one. One key property is that Gabidulin codes can be easily distinguished from random linear codes. This singular behavior has been presicely exploited by Overbeck [Ove05b, Ove05a, Ove08] to mount attacks.

Definition 4.10. For any integer $i \geqslant 0$ let $\Lambda_i : \mathcal{M}_{k,n}(\mathbb{F}_{q^m}) \longrightarrow \mathcal{M}_{ik,n}(\mathbb{F}_{q^m})$ be the $\mathbb{F}_q$-linear operator that maps any $\boldsymbol{M}$ from $\mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ to $\Lambda_i(\boldsymbol{M})$ where by definition:

$$\Lambda_i(\boldsymbol{M}) \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{M}^{[0]} \\ \vdots \\ \boldsymbol{M}^{[i]} \end{pmatrix}. \tag{4.14}$$

For any code $\mathscr{G}$ generated by a matrix $\boldsymbol{G}$ we denote by $\Lambda_i(\mathscr{G})$ the code generated by $\Lambda_i(\boldsymbol{G})$.

Proposition 4.13. *Let $\boldsymbol{g}$ be in $\mathbb{F}_{q^m}^n$ with $|\boldsymbol{g}| = n$ with $n \leqslant m$. For any integers k and i such that $k \leqslant n$ and $i \leqslant n-k-1$ we have:*

$$\Lambda_i\big(\mathscr{G}_k(\boldsymbol{g})\big) = \mathscr{G}_{k+i}(\boldsymbol{g}). \tag{4.15}$$

The importance of Λ_i becomes clear when one compares the dimension of the code spanned by $\Lambda_i(\boldsymbol{G})$ for a randomly drawn matrix $\boldsymbol{G}$ and the dimension obtained when $\boldsymbol{G}$ generates a Gabidulin code.

Proposition 4.14. *If $\mathscr{A} \subset \mathbb{F}_{q^m}^n$ is a code generated by a random matrix from $\mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ then with a high probability:*

$$\dim \Lambda_i(\mathscr{A}) = \min\big\{n, (i+1)k\big\} \tag{4.16}$$

In the case of a Gabidulin code, we get a different situation as explained by Proposition 4.13. Thus there is property that is computable in polynomial time and distinguishes a Gabidulin code from a random one. This can be used in a cryptanalysis context. In fact, Overbeck [Ove08] has proven that, for a public matrix $\boldsymbol{G}_p$ given by equation (4.12) with $\boldsymbol{X}_2 = \boldsymbol{0}$ (in particular all the entries of $\boldsymbol{P}$ belong to $\mathbb{F}_q$), it is possible (under certain conditions) to find in polynomial time an alternative decomposition of $\boldsymbol{G}_p$ of the from $\boldsymbol{S}^*(\boldsymbol{X}^* \mid \boldsymbol{G}^*)\boldsymbol{P}^*$ using the operator Λ_i. This decomposition allows to decrypt any ciphertext computed with $\boldsymbol{G}_p$.

4.2.2 Overbeck's Attack

To explain this attack, we will need the following lemma:

Lemma 4.15. *Let $\boldsymbol{P} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$ where $\boldsymbol{A}$ and $\boldsymbol{D}$ are square matrices. Then $\boldsymbol{P}$ is non singular if and only if $\boldsymbol{A}$ and $\boldsymbol{D}$ are non singular and the inverse of $\boldsymbol{P}$ is:*

$$\boldsymbol{P}^{-1} = \begin{pmatrix} \boldsymbol{A}^{-1} & \boldsymbol{0} \\ -\boldsymbol{D}^{-1}\boldsymbol{C}\boldsymbol{A}^{-1} & \boldsymbol{D}^{-1} \end{pmatrix}$$

Proof. It is clear that the non singularity of $\boldsymbol{A}$ or $\boldsymbol{D}$ implies the non singularity of $\boldsymbol{P}$. Conversely, if $\boldsymbol{P}$ is non singular, the structure of $\boldsymbol{P}$ allows to deduce that $\boldsymbol{A}$ and $\boldsymbol{D}$ are non singular. The formula of the inverse is obvious. □

Let assume that $\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{X} \mid \boldsymbol{G})\boldsymbol{P}$ is the public generator matrix that generates $\mathscr{C}_{\text{pub}}$ with $\boldsymbol{P} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$, $\boldsymbol{X} \in \mathcal{M}_{k,\ell}(\mathbb{F}_{q^m})$ and $\boldsymbol{G}$ generates a Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$ where $|\boldsymbol{g}| = n$. Observe that $\Lambda_i(\boldsymbol{G}_{\text{pub}})$ can be written as

$$\Lambda_i(\boldsymbol{G}_{\text{pub}}) \;=\; \boldsymbol{S}_{\text{ext}}\Big(\Lambda_i(\boldsymbol{X}) \mid \Lambda_i(\boldsymbol{G})\Big)\boldsymbol{P} \tag{4.17}$$

where

$$\boldsymbol{S}_{\text{ext}} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{S}^{[0]} & & \boldsymbol{0} \\ & \ddots & \\ \boldsymbol{0} & & \boldsymbol{S}^{[i]} \end{pmatrix}.$$

Since $\Lambda_i(\boldsymbol{G})$ generates $\mathscr{G}_{k+i}(\boldsymbol{g}) = \mathscr{G}_{n-1}(\boldsymbol{g})$, there exists $\boldsymbol{S}' \in \mathsf{GL}_{k(i+1)i}(\mathbb{F}_{q^m})$ such that

$$\boldsymbol{S}'\Lambda_i(\boldsymbol{G}_{\text{pub}}) = \begin{pmatrix} \boldsymbol{X}^* & \boldsymbol{G}_{n-1} \\ \boldsymbol{X}^{**} & \boldsymbol{0} \end{pmatrix} \boldsymbol{P} \tag{4.18}$$

where $\boldsymbol{X}^* \in \mathcal{M}_{(n-1),\ell}(\mathbb{F}_{q^m})$, $\boldsymbol{X}^{**} \in \mathcal{M}_{(k(i+1)-n+1),\ell}(\mathbb{F}_{q^m})$ and $\boldsymbol{G}_{n-1} \in \mathcal{M}_{(n-1),n}(\mathbb{F}_{q^m})$ generates $\mathscr{G}_{n-1}(\boldsymbol{g})$. Using (4.18), one can deduce that by taking $i = n-k-1$

$$\dim \Lambda_{n-k-1}(\mathscr{C}_{\text{pub}}) = n - 1 + \mathtt{rank}(\boldsymbol{X}^{**}).$$

In the particular case where $\mathtt{rank}(\boldsymbol{X}^{**}) = \ell$ then $\dim \Lambda_i(\mathscr{C}_{\text{pub}}) = n + \ell - 1$ and thus $\dim \Lambda_i(\mathscr{C}_{\text{pub}})^\perp = 1$. Furthermore, if $\boldsymbol{h}$ is a non zero vector from $\mathscr{G}_{n-1}(\boldsymbol{g})^\perp$ and we set $\boldsymbol{h}^* = (\boldsymbol{0} \mid \boldsymbol{h})\left(\boldsymbol{P}^{-1}\right)^T$ then under the assumption that $\mathtt{rank}(\boldsymbol{X}^{**}) = \ell$ we have

$$\Lambda_{n-k-1}(\mathscr{C}_{\text{pub}})^\perp = \mathbb{F}_{q^m}\boldsymbol{h}^*. \tag{4.19}$$

Proposition 4.16. *Let $\boldsymbol{v} \in \Lambda_{n-k-1}(\mathscr{C}_{\text{pub}})^\perp$ with $\boldsymbol{v} \neq \boldsymbol{0}$. Any matrix $\boldsymbol{T} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ that satisfies $\boldsymbol{vT} = (\boldsymbol{0} \mid \boldsymbol{h}')$ with $\boldsymbol{h}' \in \mathbb{F}_{q^m}^n$ is an alternative column scrambler matrix, that is to say, there exist $\boldsymbol{Z}$ in $\mathcal{M}_{k,\ell}(\mathbb{F}_{q^m})$ and $\boldsymbol{G}^*$ that generates a Gabidulin code $\mathscr{G}_k(\boldsymbol{g}^*)$ such that*

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{Z} \mid \boldsymbol{G}^*)\boldsymbol{T}.$$

Proof. From (4.19) there exists $\alpha \in \mathbb{F}_{q^m}$ such that $\boldsymbol{v} = \alpha\boldsymbol{h}^* = (\boldsymbol{0} \mid \alpha\boldsymbol{h})\left(\boldsymbol{P}^{-1}\right)^T$ where $\boldsymbol{h}$ is a non zero vector of $\mathscr{G}_{n-1}(\boldsymbol{g})^\perp$. Let $\boldsymbol{T} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ such that $\boldsymbol{vT}^T = (\boldsymbol{0} \mid \boldsymbol{h}')$ and consider the matrices $\boldsymbol{A} \in \mathcal{M}_{\ell,\ell}(\mathbb{F}_q)$ and $\boldsymbol{D} \in \mathcal{M}_{n,n}(\mathbb{F}_q)$ so that

$$\boldsymbol{TP}^{-1} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}.$$

We have then the following equalities

$$\tilde{\boldsymbol{h}}\boldsymbol{T}^T = (\boldsymbol{0} \mid \alpha\boldsymbol{h})\left(\boldsymbol{P}^{-1}\right)^T\boldsymbol{T}^T = (\boldsymbol{0} \mid \alpha\boldsymbol{h})\left(\boldsymbol{TP}^{-1}\right)^T = (\boldsymbol{0} \mid \boldsymbol{h}') \tag{4.20}$$

It comes out from (6.17) that $\boldsymbol{h}\boldsymbol{B}^T = \boldsymbol{0}$ and hence $\boldsymbol{B} = \boldsymbol{0}$ since $|\boldsymbol{h}| = n$. So we can write $\boldsymbol{TP}^{-1} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{0} \\ \boldsymbol{C} & \boldsymbol{D} \end{pmatrix}$ and using Lemma 4.15, $\boldsymbol{PT}^{-1} = \begin{pmatrix} \boldsymbol{A}' & \boldsymbol{0} \\ \boldsymbol{C}' & \boldsymbol{D}' \end{pmatrix}$. Consequently,

$$\boldsymbol{G}_{pub}\boldsymbol{T}^{-1} = \boldsymbol{S}(\boldsymbol{X} \mid \boldsymbol{G})\begin{pmatrix} \boldsymbol{A}' & \boldsymbol{0} \\ \boldsymbol{C}' & \boldsymbol{D}' \end{pmatrix} = \boldsymbol{S}(\boldsymbol{Z} \mid \boldsymbol{G}^*)$$

where $\boldsymbol{G}^* = \boldsymbol{G}\boldsymbol{D}'$ is a generator matrix of an $(n,k)-$Gabidulin code. So $\boldsymbol{T}$ is an alternative column scrambler matrix for the system. □

Overbeck's attack is achieves with $O\left((n+\ell)^3\right)$ operations on $\mathbb{F}_{q^m}$. Furthermore, it uses crucially two important facts: The column scrambler matrix $\boldsymbol{P}$ is defined on the based field $\mathbb{F}_q$, and the codimension of $\Lambda_{n-k-1}\left(\mathscr{A}\right)$ is 1. Several works propose to resist to Overbeck's attack either by taking special distortion matrix so that the second property is not true as in [Loi10, RGH10], or by taking a column scrambler matrix defined over the extension field $\mathbb{F}_{q^m}$ as in [Gab08, GRH09, RGH11]. Recently a new structural attacked appeared in [HMR15] which can be used as an alternative to Overbeck's attack. Especially the technique of [HMR15], consisting of looking at the elements of rank one in an appropriate code derives from the public code, allows to break the variants of [Loi10, RGH10]. Concerning the variants of [Gab08, GRH09, RGH11], no structural attack have been presented up to this thesis and the best attacks are the new generic decoding algorithms of [GRS16, HTMR16].

Chapter 5

Cryptanalysis of Recent Variants of the GPT Cryptosystem

Introduction

In this chapter, we study the security of the recent variants of the GPT cryptosystem proposed in [Gab08, GRH09, RGH11, RGH10].

The variants of [Gab08, GRH09, RGH11] consist to take a column scrambler matrix with coefficients in the extension field. We show that, it is still possible to recover a secret Gabidulin code using precisely Overbeck's technique. Our analysis shows that by applying the operator Λ_i with $i < n - k - 1$, we obtain a Gabidulin code whose error correction t^* is indeed strictly less than the error correction of the secret original Gabidiulin code but t^* is strictly greater than the number of added errors t_{pub}. In other words, an attacker is still able to decrypt any ciphertext and consequently, all the scheme presented in [Gab08, GRH09, RGH11] are actually not resistant to Overbeck's attack unlike what it was claimed by the authors. When the attack is implemented with the recommended parameters of [GRH09, RGH11], our experimental results show that the attack is very fast (less than one second). In particular, our results outperform those given in [GRS16, HTMR16] which were for a while the best attacks against the schemes of [Gab08, GRH09, RGH11]. Note that in [GRS16, HTMR16] new generic decoding algorithms that permit to attack all this variants are developed whereas our approach is directed towards recovering the structure of a Gabidulin code. We will prove that all these schemes can be broken simply with the techniques developed in [Ove08].

The other reparation from [RGH10] consists to choose an appropriate distortion matrix $\boldsymbol{X}$ so that Overbeck's attack fails. We will show that this variant of the GPT cryptosystem is equivalent to insert redundancies in the public code of a general GPT cryptosystem. We will thus show how to remove the redundancies in order to apply Overbeck's attack on the public code obtained. More precisely, puncturing the public code several times at some appropriate positions allows to get a new code on which applying the Frobenius operator appropriately allows to build an alternative secret key.

5.1 Gabidulin's General Reparation

In this section, we focus on the reparation given in [Gab08]. This paper is the first to consider a column scrambler matrix defined over the extension field. We describe only the key generation and decryption steps of the scheme since the encryption operation is not modified. To the best of our knowledge, no structural attack has been mounted against this description. The author claimed that Overbeck's attack is not applicable. But in Proposition 5.1, we prove that it is still possible to find an alternative private key using precisely Overbeck's technique.

5.1.1 Description of the Scheme

The important points are Key generation and decryption.

Key generation.

1. Pick at random $\boldsymbol{g}$ from $\mathbb{F}_{q^m}^n$ such that $|\boldsymbol{g}| = n$ and let $\boldsymbol{G}$ be a generator matrix of the Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$.

2. Pick at random $\boldsymbol{X} \in \mathcal{M}_{k,\ell}(\mathbb{F}_{q^m})$, $\boldsymbol{S}$ in $\mathsf{GL}_k(\mathbb{F}_{q^m})$ and $\boldsymbol{P}$ in $\mathsf{GL}_{n+\ell}(\mathbb{F}_{q^m})$ such that there exist $\boldsymbol{Q}_{11}$ in $\mathcal{M}_{\ell,\ell}(\mathbb{F}_{q^m})$, $\boldsymbol{Q}_{21}$ in $\mathcal{M}_{n,\ell}(\mathbb{F}_{q^m})$, $\boldsymbol{Q}_{22}$ in $\mathcal{M}_{n,n}(\mathbb{F}_q)$ and $\boldsymbol{Q}_{12}$ in $\mathcal{M}_{\ell,n}(\mathbb{F}_{q^m})$ with $|\boldsymbol{Q}_{12}| = s < t$ so that

$$\boldsymbol{P}^{-1} = \begin{pmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} \\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} \end{pmatrix}. \tag{5.1}$$

The public key is $(\boldsymbol{G}_{\text{pub}}, t_{\text{pub}})$ with $t_{\text{pub}} = t - s$ and

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{X} \mid \boldsymbol{G})\boldsymbol{P}. \tag{5.2}$$

Decryption. We have $\boldsymbol{c}\boldsymbol{P}^{-1} = \boldsymbol{m}\boldsymbol{S}(\boldsymbol{X} \mid \boldsymbol{G}) + \boldsymbol{e}\boldsymbol{P}^{-1}$. Suppose that $\boldsymbol{e} = (\boldsymbol{e}_1 \mid \boldsymbol{e}_2)$ where $\boldsymbol{e}_1 \in \mathbb{F}_{q^m}^{\ell}$ and $\boldsymbol{e}_2 \in \mathbb{F}_{q^m}^{n}$. We have:

$$\boldsymbol{e}\boldsymbol{P}^{-1} = (\boldsymbol{e}_1\boldsymbol{Q}_{11} + \boldsymbol{e}_2\boldsymbol{Q}_{21} \mid \boldsymbol{e}_1\boldsymbol{Q}_{12} + \boldsymbol{e}_2\boldsymbol{Q}_{22}) \tag{5.3}$$

It is clear that

$$|\boldsymbol{e}_1\boldsymbol{Q}_{12} + \boldsymbol{e}_2\boldsymbol{Q}_{22}| \leqslant |\boldsymbol{e}_1\boldsymbol{Q}_{12}| + |\boldsymbol{e}_2\boldsymbol{Q}_{22}| \leqslant s + t - s.$$

So the plaintext $\boldsymbol{m}$ is recovered by applying the decoding algorithm only to the last n components of $\boldsymbol{c}\boldsymbol{P}^{-1}$.

5.1.2 Cryptanalysis

We state our main result proving that Overbeck's attack is still successful by considering this time the dual of $\Lambda_i(\boldsymbol{G}_{\text{pub}})$ with $i = n - s - k - 1$.

Proposition 5.1. *There exist $\boldsymbol{X}^* \in \mathcal{M}_{k,(\ell+s)}(\mathbb{F}_{q^m})$, $\boldsymbol{P}^* \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ and a generator matrix $\boldsymbol{G}^*$ that defines an $(n-s,k)-$Gabidulin code $\mathscr{G}_k(\boldsymbol{g}^*)$ such that*

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{X}^* \mid \boldsymbol{G}^*)\boldsymbol{P}^*. \tag{5.4}$$

Furthermore, the error correction capability t^* of $\mathscr{G}_k(\boldsymbol{g}^*)$ is equal to $t - \frac{1}{2}s$, and hence $t^* > t_{\text{pub}}$.

The proof of this proposition requires to prove the following lemma.

Lemma 5.2. *There exist $\boldsymbol{P}_{11}$ in $\mathsf{GL}_{\ell+s}(\mathbb{F}_{q^m})$, $\boldsymbol{P}_{21}$ in $\mathcal{M}_{(n-s),(\ell+s)}(\mathbb{F}_{q^m})$ and $\boldsymbol{P}_{22}$ in $\mathsf{GL}_{n-s}(\mathbb{F}_q)$ such that*

$$\boldsymbol{P} = \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L} \end{pmatrix} \begin{pmatrix} \boldsymbol{P}_{11} & \boldsymbol{0} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{pmatrix} \tag{5.5}$$

with $\boldsymbol{L}$ and $\boldsymbol{R}$ belonging to $\mathsf{GL}_n(\mathbb{F}_q)$.

Proof. By assumption $|\boldsymbol{Q}_{12}| = s < t$ so there exist $\boldsymbol{R}$ in $\mathsf{GL}_n(\mathbb{F}_q)$ and $\boldsymbol{Q}'_{12}$ in $\mathcal{M}_{\ell,s}(\mathbb{F}_{q^m})$ such that $\boldsymbol{Q}_{12}\boldsymbol{R} = (\boldsymbol{Q}'_{12} \mid \boldsymbol{0})$. We set $\boldsymbol{Q}_{22}\boldsymbol{R} = (\boldsymbol{Q}'_{22} \mid \boldsymbol{Q}'_{23})$ where $\boldsymbol{Q}'_{22}$ in $\mathcal{M}_{n,s}(\mathbb{F}_q)$ and $\boldsymbol{Q}'_{23}$ in $\mathcal{M}_{n,(n-s)}(\mathbb{F}_q)$. Note that we necessarily have $|\boldsymbol{Q}'_{23}| \leqslant n - s$ and therefore there exists $\boldsymbol{L} \in \mathsf{GL}_n(\mathbb{F}_q)$ such that $\boldsymbol{L}\boldsymbol{Q}'_{23} = \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{Q}''_{23} \end{pmatrix}$ with $\boldsymbol{Q}''_{23} \in \mathcal{M}_{(n-s),(n-s)}(\mathbb{F}_q)$. Thus one can rewrite

$$\begin{aligned} \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L} \end{pmatrix} \boldsymbol{P}^{-1} \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{pmatrix} &= \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L} \end{pmatrix} \begin{pmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} \\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} \end{pmatrix} \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{pmatrix} && (5.6) \\ &= \begin{pmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}'_{12} & \boldsymbol{0} \\ \boldsymbol{L}\boldsymbol{Q}_{21} & \boldsymbol{L}\boldsymbol{Q}'_{22} & \boldsymbol{L}\boldsymbol{Q}'_{23} \end{pmatrix} && (5.7) \end{aligned}$$

Observe that there exist $\boldsymbol{Q}''_{11}$ in $\mathcal{M}_{(\ell+s),(\ell+s)}(\mathbb{F}_{q^m})$ and $\boldsymbol{Q}''_{21}$ in $\mathcal{M}_{(n-s),(\ell+s)}(\mathbb{F}_{q^m})$ so that we can write

$$\begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L} \end{pmatrix} \boldsymbol{P}^{-1} \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{pmatrix} = \begin{pmatrix} \boldsymbol{Q}''_{11} & \boldsymbol{0} \\ \boldsymbol{Q}''_{21} & \boldsymbol{Q}''_{23} \end{pmatrix}.$$

Note that $\boldsymbol{Q}''_{23}$ and $\boldsymbol{Q}''_{11}$ are necessarily invertible and thanks to Lemma 4.15 the proof can be terminated. □

Remark 5.1. The proof of Lemma 5.2 is still true if it is assumed that $|\boldsymbol{Q}_{12}| < s$, and note that by construction s is necessarily less than or equal to ℓ.

We are now able to give a proof of Proposition 5.1.

Proposition 5.1. We keep the same notation as those of Lemma 5.2. Let us rewrite $\boldsymbol{GL}$ as $(\boldsymbol{G}'_1 \mid \boldsymbol{G}'_2)$ where $\boldsymbol{G}'_1$ in $\mathcal{M}_{k,s}(\mathbb{F}_{q^m})$ and $\boldsymbol{G}'_2$ in $\mathcal{M}_{k,(n-s)}(\mathbb{F}_{q^m})$ and set now $\boldsymbol{Y} = (\boldsymbol{X} \mid \boldsymbol{G}'_1)$. Observe that $\boldsymbol{G}'_2$ generates an $(n-s,k)-$Gabidulin code. We then have

$$(\boldsymbol{X} \mid \boldsymbol{G}) \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{L} \end{pmatrix} \begin{pmatrix} \boldsymbol{P}_{11} & \boldsymbol{0} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} \end{pmatrix} = (\boldsymbol{Y} \mid \boldsymbol{G}'_2) \begin{pmatrix} \boldsymbol{P}_{11} & \boldsymbol{0} \\ \boldsymbol{P}_{21} & \boldsymbol{P}_{22} \end{pmatrix} = (\boldsymbol{X}^* \mid \boldsymbol{G}^*)$$

where $\boldsymbol{X}^* = \boldsymbol{Y}\boldsymbol{P}_{11} + \boldsymbol{G}'_2\boldsymbol{P}_{21}$ and $\boldsymbol{G}^* = \boldsymbol{G}'_2\boldsymbol{P}_{22}$ is a generator matrix of an $(n-s,k)-$Gabidulin code. Hence if we set $\boldsymbol{P}^* = \begin{pmatrix} \boldsymbol{I}_\ell & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{R} \end{pmatrix}$ we then have rewritten $\boldsymbol{G}_{\text{pub}}$ as expected in (5.4). Lastly remark that $t_{\text{pub}} = t - s$ and $t^* = \frac{1}{2}(n-s-k) = \frac{1}{2}(n-k) - \frac{1}{2}s > t - s$. □

Form this proposition, it is clear that this variant can be broken by using Overbeck's attack with $i = n - s - k - 1$.

5.2 Gabidulin, Rashwan and Honary Variant

In [GRH09, RGH11] Gabidulin, Rashwan and Honary also proposed an other variant where the column scrambler has its entries defined on the extension field. This variant can be described as follows:

5.2.1 Description

We just describe the key generation and the decryption phases.

Key generation.

1. Pick at random $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ such that $|\boldsymbol{g}| = n$ and let $\boldsymbol{G} \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ be a generator matrix of the Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$. Let t_{pub} be an integer $< t$ and set $a \stackrel{\text{def}}{=} t - t_{\text{pub}}$.

2. Pick at random $\boldsymbol{S}$ in $\mathsf{GL}_k(\mathbb{F}_{q^m})$ and $\boldsymbol{P} \in \mathsf{GL}_n(\mathbb{F}_{q^m})$ such that

$$\boldsymbol{P}^{-1} = (\boldsymbol{Q}_1 \mid \boldsymbol{Q}_2) \tag{5.8}$$

where $\boldsymbol{Q}_1 \in \mathcal{M}_{n,a}(\mathbb{F}_{q^m})$ while $\boldsymbol{Q}_2 \in \mathcal{M}_{n,(n-a)}(\mathbb{F}_q)$ with $t = \frac{1}{2}(n-k)$ and $t_{\text{pub}} < t$. The public key is $(\boldsymbol{G}_{\text{pub}}, t_{\text{pub}})$ with

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{SGP}. \tag{5.9}$$

Decryption. First, we have $\boldsymbol{cP}^{-1} = \boldsymbol{mSG} + \boldsymbol{eP}^{-1}$ and $\boldsymbol{eP}^{-1} = (\boldsymbol{eQ}_1 \mid \boldsymbol{eQ}_2)$. Observe that $|\boldsymbol{eQ}_1| \leqslant a$ and $|\boldsymbol{eQ}_2| \leqslant |\boldsymbol{e}| \leqslant t_{\text{pub}}$, and since $a = t - t_{\text{pub}}$ we hence have

$$\left|\boldsymbol{eP}^{-1}\right| \leqslant |\boldsymbol{eQ}_1| + |\boldsymbol{eQ}_2| \leqslant t.$$

5.2.2 Cryptanalysis

We now prove that Overbeck's attack is still successful by considering for this scheme the dual of $\Lambda_i(\boldsymbol{G}_{\text{pub}})$ with $i = n-a-k-1$. We first introduce the matrices $\boldsymbol{Q}_{11} \in \mathcal{M}_{a,a}(\mathbb{F}_{q^m})$, $\boldsymbol{Q}_{21} \in \mathcal{M}_{n-a,a}(\mathbb{F}_{q^m})$, $\boldsymbol{Q}_{12} \in \mathcal{M}_{a,n-a}(\mathbb{F}_q)$ and $\boldsymbol{Q}_{22} \in \mathcal{M}_{n-a,n-a}(\mathbb{F}_q)$ such that

$$\boldsymbol{P}^{-1} = \begin{pmatrix} \boldsymbol{Q}_{11} & \boldsymbol{Q}_{12} \\ \boldsymbol{Q}_{21} & \boldsymbol{Q}_{22} \end{pmatrix}. \tag{5.10}$$

Note that $|\boldsymbol{Q}_{12}| \leqslant a < t$. Furthermore, by looking at the proof of Lemma 5.2, we can see that this lemma and Proposition 5.1 are still true even if $|\boldsymbol{Q}_{12}| \leqslant s$. Hence, the scheme given in [GRH09, RGH11] is nothing else but a special case of [Gab08] where $\boldsymbol{X} = \boldsymbol{0}$ and $\boldsymbol{Q}_{12}$ has all its entries in the base field $\mathbb{F}_q$. We have therefore the following corollary.

Corollary 5.3. *There exist* $\boldsymbol{P}^* \in \mathsf{GL}_n(\mathbb{F}_q)$ *and* $\boldsymbol{X} \in \mathcal{M}_{k,a}(\mathbb{F}_{q^m})$ *such that*

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{X} \mid \boldsymbol{G}^*)\boldsymbol{P}^* \tag{5.11}$$

where $\boldsymbol{G}^*$ *is a generator matrix of an* $(n-a,k)-$*Gabidulin code whose error correction capability* t^* *is equal to* $\lfloor \frac{1}{2}(t+t_{\text{pub}}) \rfloor$*, and hence* $t^* > t_{\text{pub}}$*.*

Proof. Apply Proposition 5.1 with $\ell = 0$ and $s = a$. Note that the error correction capability t^* of the code $\boldsymbol{G}^*$ is equal to $\frac{1}{2}(n-a-k)$ that is to say

$$t^* = t - \frac{1}{2}(t - t_{\text{pub}}) = \frac{1}{2}(t + t_{\text{pub}}) > t_{\text{pub}}.$$

□

We summarised in Table 5.1 our experimental results obtained with Magma V2.21-6. We give the time to find an alternative column scrambler matrix for each parameter proposed by the authors in [GRH09] and [RGH11]. In particular, our results outperform those given in [GRS16, HTMR16].

m	k	t	t_{pub}	Time (second)
20	10	5	4	$\leqslant 1$
28	14	7	3	$\leqslant 1$
28	14	7	4	$\leqslant 1$
28	14	7	5	$\leqslant 1$
28	14	7	6	$\leqslant 1$
20	10	5	4	$\leqslant 1$

Table 5.1 – Parameters from [GRH09, RGH11] where $n = m$ and at least 80-bit security.

5.3 Discussion on a More General Column Scrambler

In [GRH09] the authors proposed to reinforce the security by taking a more general column scrambler matrix of the form $\boldsymbol{TP}$ where $\boldsymbol{T}$ is an invertible matrix with its entries in $\mathbb{F}_q$ and $\boldsymbol{P}$ is defined over the extension field as it is done in [Gab08, GRH09, RGH10]. We shall consider Gabidulin's general reparation [Gab08] since [GRH09, RGH10] are particular cases but we emphasize that this new protection was only defined in [GRH09, RGH10]. Assuming that $\boldsymbol{P}$ is then as in (5.1), the public key is then of the form

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}\left(\boldsymbol{X} \mid \boldsymbol{G}\right)\boldsymbol{TP}. \tag{5.12}$$

The decryption of a ciphertext $\boldsymbol{c}$ starts by calculating $\boldsymbol{cP}^{-1}\boldsymbol{T}^{-1} = \boldsymbol{mS}\left(\boldsymbol{X} \mid \boldsymbol{G}\right) + \boldsymbol{eP}^{-1}\boldsymbol{T}^{-1}$ where $\boldsymbol{e}$ is of rank weight t_{pub} and $s = |\boldsymbol{Q}_{12}|$. The retrieving of the original plaintext $\boldsymbol{m}$ is possible provided that $t_{\text{pub}} = t - \ell - s$ because $\boldsymbol{cP}^{-1}\boldsymbol{T}^{-1} = \boldsymbol{mS}\left(\boldsymbol{X} \mid \boldsymbol{G}\right) + \boldsymbol{eP}^{-1}\boldsymbol{T}^{-1}$. Suppose that $\boldsymbol{e} = (\boldsymbol{e}_1 \mid \boldsymbol{e}_2)$ where $\boldsymbol{e}_1 \in \mathbb{F}_{q^m}^{\ell}$ and $\boldsymbol{e}_2 \in \mathbb{F}_{q^m}^{n}$, then we also have

$$\boldsymbol{eP}^{-1}\boldsymbol{T}^{-1} = \left(\boldsymbol{e}_1\boldsymbol{Q}_{11} + \boldsymbol{e}_2\boldsymbol{Q}_{21} \mid \boldsymbol{e}_1\boldsymbol{Q}_{12} + \boldsymbol{e}_2\boldsymbol{Q}_{22}\right)\boldsymbol{T}^{-1}. \tag{5.13}$$

It is clear that $|\boldsymbol{e}_1\boldsymbol{Q}_{12} + \boldsymbol{e}_2\boldsymbol{Q}_{22}| \leqslant |\boldsymbol{e}_1\boldsymbol{Q}_{12}| + |\boldsymbol{e}_2\boldsymbol{Q}_{22}| \leqslant s + t_{\text{pub}}$ and hence it implies that

$$\left|\boldsymbol{eP}^{-1}\boldsymbol{T}^{-1}\right| = \left|\boldsymbol{eP}^{-1}\right| \leqslant |\boldsymbol{e}_1\boldsymbol{Q}_{11} + \boldsymbol{e}_2\boldsymbol{Q}_{21}| + |\boldsymbol{e}_1\boldsymbol{Q}_{12} + \boldsymbol{e}_2\boldsymbol{Q}_{22}| \leqslant \ell + s + t_{\text{pub}}.$$

Therefore the plaintext $\boldsymbol{m}$ is recovered by applying the decoding algorithm only to the last n components of $\boldsymbol{cP}^{-1}\boldsymbol{T}^{-1}$. But in this case, the rank weight of the last n components of $\boldsymbol{eP}^{-1}\boldsymbol{T}^{-1}$ is not necessarily less than or equal to $t_{\text{pub}} + s$ but rather to $t_{\text{pub}} + s + \ell$. Consequently, the decryption will always succeed if it assumed that $t_{\text{pub}} = t - s - \ell$ otherwise the decoding may fail. Hence, we see why this new reparation was just proposed for the case where $\ell = 0$ *i.e.* without any distortion matrix since otherwise its deteriorates the performances of the original scheme.

We now study more precisely the security this protection might bring in for the general scheme of [Gab08]. First, rewrite $\boldsymbol{T}$ as

$$\boldsymbol{T} = \begin{pmatrix} \boldsymbol{T}_{11} & \boldsymbol{T}_{12} \\ \boldsymbol{T}_{21} & \boldsymbol{T}_{22} \end{pmatrix} \tag{5.14}$$

where $\boldsymbol{T}_{11} \in \mathcal{M}_{\ell,\ell}\left(\mathbb{F}_q\right)$, $\boldsymbol{T}_{21} \in \mathcal{M}_{n,\ell}\left(\mathbb{F}_q\right)$, $\boldsymbol{T}_{12} \in \mathcal{M}_{\ell,n}\left(\mathbb{F}_q\right)$ and $\boldsymbol{T}_{22} \in \mathcal{M}_{n,n}\left(\mathbb{F}_q\right)$. On the other hand, by Lemma 5.2 the matrix $\boldsymbol{P}$ can be expressed as (5.5). We can find then $\boldsymbol{X}_1$ in $\mathcal{M}_{k,(\ell+s)}\left(\mathbb{F}_{q^m}\right)$ and $\boldsymbol{X}_2$ in $\mathcal{M}_{k,(n-s)}\left(\mathbb{F}_{q^m}\right)$ such that

$$\left(\boldsymbol{X} \mid \boldsymbol{G}\right)\boldsymbol{TP} \;=\; \left(\boldsymbol{X}_1 \mid \boldsymbol{X}_2 + \boldsymbol{G}_1^*\right)$$

where $\boldsymbol{G}_1^*$ generates an $(n-s,k)-$Gabidulin code and $|\boldsymbol{X}_2| = |\boldsymbol{X}| \leqslant \ell$. From Proposition 4.12 and by taking $t_2 = \ell$, there exist $\boldsymbol{P}^* \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$, $\boldsymbol{X}^* \in \mathcal{M}_{k,(2\ell+s)}\left(\mathbb{F}_{q^m}\right)$ and $\boldsymbol{G}^*$ that generates an $(n-s-\ell,k)-$Gabidulin code such that

$$\left(\boldsymbol{X} \mid \boldsymbol{G}\right)\boldsymbol{TP} = \left(\boldsymbol{X}^* \mid \boldsymbol{G}^*\right)\boldsymbol{P}^*. \tag{5.15}$$

We have therefore proven the following proposition.

Proposition 5.4. *Assume that* $\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}\,(\boldsymbol{X} \mid \boldsymbol{G})\,\boldsymbol{T}\boldsymbol{P}$ *where* $\boldsymbol{T} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ *and* $\boldsymbol{G}$ *has the form* (5.1). *Then, there exist* $\boldsymbol{P}^*$ *in* $\mathsf{GL}_{n+\ell}(\mathbb{F}_q)$, $\boldsymbol{X}^*$ *in* $\mathcal{M}_{k,(2\ell+s)}\,(\mathbb{F}_{q^m})$ *and a matrix* $\boldsymbol{G}^*$ *that generates an* $(n-s-\ell,k)-$*Gabidulin code* $\mathscr{G}_k\,(\boldsymbol{g}^*)$ *such that*

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{X}^* \mid \boldsymbol{G}^*)\boldsymbol{P}^*.$$

Furthermore, the correction capability t^* *of* $\mathscr{G}_k\,(\boldsymbol{g}^*)$ *is greater than* $t - \frac{1}{2}(\ell+s)$. *In particular* $t^* > t_{\text{pub}}$.

This result shows that actually this new proposed protection does not improve the security even when applied with the scheme for which a distortion matrix $\boldsymbol{X}$ is used. An example where this protection was used and turns out to be useless is the scheme given in [GRH09, RGH11].

Related construction In [GP13, GP14], another variant is also proposed. This variant consists to use a column scrambler matrix $\boldsymbol{P}$ such that

$$\boldsymbol{P}^{-1} = \boldsymbol{T} + \boldsymbol{Z} \tag{5.16}$$

$\boldsymbol{T} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ and $\boldsymbol{Z} \in \mathcal{M}_{n+\ell,n+\ell}\,(\mathbb{F}_{q^m})$ with $|\boldsymbol{Z}| = s$. However, this last variant was shown in [UG14] to be equivalent to the general GPT cryptosystem [GO01] and hence not secure. The following proposition compares the variant of [GP13, GP14] with the variant of section 5.2.

Proposition 5.5. *The matrix* $\boldsymbol{P}$ *can be written as*

$$\boldsymbol{P} = \boldsymbol{P}^*\boldsymbol{Q}$$

with $\boldsymbol{P}^* \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ *and* $\boldsymbol{Q}^{-1} = (\boldsymbol{Q}_1 \mid \boldsymbol{Q}_2) \in \mathsf{GL}_{n+\ell}(\mathbb{F}_{q^m})$ *such that* $\boldsymbol{Q}_1 \in \mathcal{M}_{n+\ell,s}\,(\mathbb{F}_{q^m})$ *and* $\boldsymbol{Q}_2 \in \mathcal{M}_{n+\ell,n+\ell-s}\,(\mathbb{F}_q)$.

Proof. Since $\boldsymbol{P}^{-1} = \boldsymbol{T} + \boldsymbol{Z}$ with $|\boldsymbol{Z}| = s$, there exists $\boldsymbol{R} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ and $\boldsymbol{Z}^* \in \mathcal{M}_{n+\ell,s}\,(\mathbb{F}_{q^m})$ such that $\boldsymbol{Z} = (\boldsymbol{Z}^* \mid \boldsymbol{0})\,\boldsymbol{R}$. Letting $\boldsymbol{T}\boldsymbol{R}^{-1} = (\boldsymbol{T}_1 \mid \boldsymbol{T}_2)$ we have:

$$\boldsymbol{P}^{-1} = \boldsymbol{T} + (\boldsymbol{Z}^* \mid \boldsymbol{0})\,\boldsymbol{R} = \left[\boldsymbol{T}\boldsymbol{R}^{-1} + (\boldsymbol{Z}^* \mid \boldsymbol{0})\right]\boldsymbol{R} = (\boldsymbol{T}_1 + \boldsymbol{Z}^* \mid \boldsymbol{T}_2)\,\boldsymbol{R}$$

Taking $\boldsymbol{P}^* = \boldsymbol{R}^{-1}$ and $\boldsymbol{Q} = (\boldsymbol{T}_1 + \boldsymbol{Z}^* \mid \boldsymbol{T}_2)^{-1}$ achieves the proof. □

We understand that the scheme given in [GP13, GP14] is nothing else but a special case of [GRH09, RGH11] this implies that this scheme is not secure and also confirms the result of [UG14].

5.4 The Smart Approach of the GPT Cryptosystem

In [RGH10], another way to avoid structural attacks on the GPT cryptosystem was proposed. It consists to choose an appropriate distortion matrix $\boldsymbol{X}$. A first structural attack on this variant was proposed in [HMR15]. In this section, we describe this reparation and we give a new and very simple algorithm that recovers an alternative secret key in polynomial time. This attack is related to the attack presented in Chapter 3 in a hamming metric context.

5.4.1 Description

The only difference is on the generation of $\boldsymbol{X}$. The authors proposed to take $\boldsymbol{X} \in \mathcal{M}_{k,\ell}(\mathbb{F}_{q^m})$ that is a concatenation of a $q-$Vandermonde matrix $\boldsymbol{X}_1 \in \mathcal{M}_{k,a}(\mathbb{F}_{q^m})$ and a random matrix $\boldsymbol{X}_2 \in \mathcal{M}_{k,\ell-a}(\mathbb{F}_{q^m})$ with $0 < a < \ell$.

5.4.2 Cryptanalysis

Let $\boldsymbol{S} \in \mathsf{GL}_k(\mathbb{F}_{q^m})$, $\boldsymbol{X}_2 \in \mathcal{M}_{k,\ell-a}(\mathbb{F}_{q^m})$, $\boldsymbol{b} = (b_1, \cdots, b_a)$ and

$$\boldsymbol{X}_1 = \begin{pmatrix} b_1^{[0]} & \cdots & b_a^{[0]} \\ \vdots & & \vdots \\ b_1^{[k-1]} & \cdots & b_a^{[k-1]} \end{pmatrix}. \tag{5.17}$$

We have $\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{X}_1 \mid \boldsymbol{X}_2 \mid \boldsymbol{G})\boldsymbol{P}$ with $\boldsymbol{P} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$. We start the cryptanalysis by the following lemma:

Lemma 5.6. *There exists $\boldsymbol{P}^* \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ and $\boldsymbol{G}^* \in \mathcal{M}_{k,n+s}(\mathbb{F}_{q^m})$ a generator matrix of a Gabidulin code such that*

$$\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}(\boldsymbol{0} \mid \boldsymbol{X}_2 \mid \boldsymbol{G}^*)\boldsymbol{P}^*$$

s being an integer verifying $0 \leqslant s \leqslant a$ and $n + s \leqslant m$.

Proof. Let $\boldsymbol{g}' = (\boldsymbol{b} \mid \boldsymbol{g}) \in \mathbb{F}_{q^m}^{a+n}$. Since $|\boldsymbol{g}'| \geqslant |\boldsymbol{g}| = n$, let s be an integer such that $|\boldsymbol{g}'| = n + s$. Clearly, we have $s \leqslant a$ and $|(\boldsymbol{X}_1 \mid \boldsymbol{G})| = |\boldsymbol{g}'| = n + s$. So there exists a matrix $\boldsymbol{Q} \in \mathsf{GL}_{n+a}(\mathbb{F}_q)$ such that $(\boldsymbol{X}_1 \mid \boldsymbol{G})\boldsymbol{Q} = (\boldsymbol{0} \mid \boldsymbol{G}^*)$ where $\boldsymbol{G}^* \in \mathcal{M}_{k,n+s}(\mathbb{F}_{q^m})$ is a generator matrix of a Gabidulin code $\mathscr{G}_k(\boldsymbol{g}^*)$ with $\boldsymbol{g}'\boldsymbol{Q} = (\boldsymbol{0} \mid \boldsymbol{g}^*)$. This implies that there exists a matrix $\boldsymbol{R} \in \mathsf{GL}_{n+\ell}(\mathbb{F}_q)$ such that $(\boldsymbol{X}_1 \mid \boldsymbol{X}_2 \mid \boldsymbol{G})\boldsymbol{R} = (\boldsymbol{0} \mid \boldsymbol{X}_2 \mid \boldsymbol{G}^*)$. To finish the proof, take $\boldsymbol{P}^* = \boldsymbol{R}^{-1}\boldsymbol{P}$. Since $\boldsymbol{g}'$ belongs to $\mathbb{F}_{q^m}^{\ell+n}$ we have $|\boldsymbol{g}'| = n + s \leqslant m$. □

Let $\mathscr{C}_{\text{pub}}$ be the code generated by $\boldsymbol{G}_{\text{pub}}$. We then have the following proposition:

Proposition 5.7. *The code $\mathscr{C}_{\text{pub}}$ is the public code of a general GPT cryptosystem with $w = a - s$ redundancies.*

Proof. We have $\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}\,(\boldsymbol{0} \mid \boldsymbol{X}_2 \mid \boldsymbol{G}^*)\,\boldsymbol{P}^*$. Let us suppose that $\boldsymbol{P}^* = \begin{pmatrix} \boldsymbol{Q}_1 \\ \boldsymbol{Q}_2 \end{pmatrix}$ with $\boldsymbol{Q}_1 \in \mathcal{M}_{w,n+\ell}(\mathbb{F}_q)$ and $\boldsymbol{Q}_2 \in \mathcal{M}_{n+\ell-w,n+\ell}(\mathbb{F}_q)$. We have $\boldsymbol{G}_{\text{pub}} = \boldsymbol{S}\,(\boldsymbol{X}_2 \mid \boldsymbol{G}^*)\,\boldsymbol{Q}_2$ and $\texttt{rank}(\boldsymbol{Q}_2) = n + \ell - w$. Let us suppose without loss of generality that the matrix $\boldsymbol{Q}_2^*$ of the first $n + \ell - w$ columns of $\boldsymbol{Q}_2$ is of full rank. Let $\boldsymbol{G}_{\text{pub}}^* = \boldsymbol{S}\,(\boldsymbol{X}_2 \mid \boldsymbol{G}^*)\,\boldsymbol{Q}_2^*$ and $\boldsymbol{X} = \boldsymbol{S}\,(\boldsymbol{X}_2 \mid \boldsymbol{G}^*)\,\boldsymbol{Q}_2^{**}$ where $\boldsymbol{Q}_2^{**}$ is the last w columns of $\boldsymbol{Q}_2$. Then $\boldsymbol{G}_{\text{pub}} = \left(\boldsymbol{G}_{\text{pub}}^* \mid \boldsymbol{X}\right)$. One can remark to finish that $\boldsymbol{G}_{\text{pub}}^*$ is a generator matrix of a general GPT cryptosystem. □

From the above proposition, if the w positions of redundancy are identified and removed, a cryptanalysis can use Overbeck's attack to build an alternative secret key. In the sequel we show how to know that a column of $\boldsymbol{G}_{\text{pub}}$ can be remove or not. By the above proof, we can remark that a set $\boldsymbol{I} = \{i_1, ..., i_w\} \subset \{1, 2, ..., n + \ell\}$ can be considered as the set of positions of redundancy if and only if by removing all the corresponding columns in $\boldsymbol{Q}_2$, we get a square matrix of full rank. Let $f = n+s-k$, $\Lambda_f(\mathscr{C}_{\text{pub}})$ the code generated by $\Lambda_f(\boldsymbol{G}_{\text{pub}})$, $i \in \{1, 2, ..., n+\ell\}$ and $\mathscr{C}_{\text{pub}}^i$ the punctured code of $\mathscr{C}_{\text{pub}}$ at position i. We then have the following proposition:

Proposition 5.8. *The position i can be considered as a redundancy position if and only if*

$$\dim \Lambda_f(\mathscr{C}_{\text{pub}}^i) = n + s + \ell - a$$

Proof. Let $\boldsymbol{Q}_2^i$ be the matrix obtained from $\boldsymbol{Q}_2$ by removing the i^{th} column and $\boldsymbol{G}_{\text{pub}}^i$ be the matrix obtained from $\boldsymbol{G}_{\text{pub}}$ by removing the i^{th} column. We have:

$$\dim \Lambda_f(\mathscr{C}_{\text{pub}}^i) = \texttt{rank}(\Lambda_f(\boldsymbol{G}_{\text{pub}}^i)) = \texttt{rank}(\Lambda_f\,(\boldsymbol{X}_2 \mid \boldsymbol{G}^*)\,\boldsymbol{Q}_2^i)$$

Since $\boldsymbol{X}_2$ is a random matrix, with a high probability we have

$$\dim \Lambda_f(\mathscr{C}_{\text{pub}}^i) = \min\{\texttt{rank}(\Lambda_f\,(\boldsymbol{X}_2 \mid \boldsymbol{G}^*)), \texttt{rank}(\boldsymbol{Q}_2^i)\} = \min\{n+s+\ell-a, \texttt{rank}(\boldsymbol{Q}_2^i)\}$$

If i can be consider as a position of redundancy we will have $\texttt{rank}(\boldsymbol{Q}_2^i) = \texttt{rank}(\boldsymbol{Q}_2) = n + s + \ell - a$ and $\dim \Lambda_f(\mathscr{C}_{\text{pub}}^i) = n + s + \ell - a$. Else we will have $\texttt{rank}(\boldsymbol{Q}_2^i) = n + s + \ell - a - 1$ and $\dim \Lambda_f(\mathscr{C}_{\text{pub}}^i) = n + s + \ell - a - 1$. □

It is easy for an adversary to use the previous proposition to identify a set $\boldsymbol{I}$ of w positions of redundancy. To fully break the system, one can apply Overbeck's attack with $f = n + s - k - 1$, but the value of s is not known. For the case $m = n$, it is easy to see thanks to Lemma 5.6 that s is equal to 0 and in a general context ($n \leqslant m$), one can remark from the same lemma that the integer s is the smallest one that satisfies

$$\texttt{rank}(\Lambda_{n+s-k}(\boldsymbol{G}_{\text{pub}})) = \texttt{rank}(\Lambda_{n+s+1-k}(\boldsymbol{G}_{\text{pub}}))$$

We summarise the attack in Algorithm 2.

Algorithm 2 Key Recovery of the Smart Approach of the GPT Cryptosystem

$s \leftarrow a$
while $\mathtt{rank}(\Lambda_{n+s-k}(\boldsymbol{G}_{\text{pub}})) = \mathtt{rank}(\Lambda_{n+s+1-k}(\boldsymbol{G}_{\text{pub}}))$ **do**
    $s \leftarrow s - 1$
end while
$s \leftarrow s + 1$
$w \leftarrow a - s$
$y \leftarrow n + s + \ell - a$
$f \leftarrow n + s - k$
$Z \leftarrow \{1, ..., Length(\mathscr{C}_{\text{pub}})\}$ and $J \leftarrow [\,]$
$j \leftarrow Random(Z)$
while $\sharp J \neq w$ **do**
    if $\dim(\Lambda_{n+s-k}(\mathscr{C}^j_{\text{pub}})) = y$ **then**
        $J \leftarrow HorizontalJoin(J, [j])$
        $\mathscr{C}_{\text{pub}} \leftarrow \mathscr{C}^j_{\text{pub}}$
        $Z \leftarrow \{1, ..., Length(\mathscr{C}_{\text{pub}})\}$
        $j \leftarrow Random(Z)$
    else
        $Z \leftarrow Z \setminus \{j\}$
        $j \leftarrow Random(Z)$
    end if
end while
return $\mathscr{C}_{\text{pub}}$, J
Define $\boldsymbol{G}_{\text{pub}}$ as the generator matrix of $\mathscr{C}_{\text{pub}}$
Apply Overbeck's algorithm on $\boldsymbol{G}_{\text{pub}}$ with $f = n + s - k - 1$

Complexity and Experimental Results

During the computation phase of s, the main computations are $\mathtt{rank}(\Lambda_{n+s-k}(\boldsymbol{G}_{\text{pub}}))$ and $\mathtt{rank}(\Lambda_{n+s+1-k}(\boldsymbol{G}_{\text{pub}}))$ which are computed at most a times with a complexity $O(a(n+\ell)^3)$. To identify a set of $w = a - s$ random redundancies, the main computation is $\dim(\Lambda_{n+s-k}(\mathscr{C}^j_{\text{pub}}))$ (for $j \in \{1, ..., n+\ell\}$) which is done at most $n + \ell$ times. So the complexity of this step is $O((n+\ell)^4)$. By considering the final step that consists to apply Overbeck's attack, the overall complexity is $O((n+\ell)^4)$ operations on $\mathbb{F}_{q^m}$ since the complexity of this final step is $O((n+\ell)^3)$ operations on $\mathbb{F}_{q^m}$. We implemented the attack (for $m \leqslant 30$ and for several values of a such that $am \geqslant 60$ as proposed in [RGH10]) on Magma V2.21-6 and a secret key was always found in less than 5 seconds. This confirms the efficiency of the approach.

Conclusion

The apparition of Overbeck's attack prompted some authors to invent reparations to hide more the structure of the Gabidulin codes. One trend advocated the use of

a right column scrambler with entries in the extension field as it is done in [Gab08, GRH09, RGH11]. Our analysis shows that these reparations aiming at resisting Overbeck's structural attack do fail precisely against it. By applying appropriately Overbeck's technique, we were able to construct a Gabidulin code that has the same dimension as the original one but with a lower length. Hence, we obtain a degraded Gabidulin code in terms of error correction capabilities but we prove that the degradation does not forbid the error correction of any ciphertext. Furthermore, when the attack is implemented, the practical results we obtained outperform those given in [GRS16, HTMR16] which were up to our paper the best attacks against the schemes of [Gab08, GRH09, RGH11]. We also considered in Section 5.3 the case where an isometric transformation is applied in conjunction with a right column scrambler which has its entries in the extension field. We proved that this protection is useless both in terms of performance and security.

The other kind of reparation is followed by the series of works in [Loi10, RGH10] which propose to resist to Overbeck's attack by taking a distorsion matrix $\boldsymbol{X}$ so that the codimension of $\Lambda_{n-k-1}(\mathscr{A})$ is equal to a where a is sufficiently large to prevent an exhaustive search. But these reparations were also cryptanalyzed in [GRS16, HTMR16] and recently by a new approach in [HMR15]. We have also shown that the variant of [RGH10] can be seen like a general GPT Cryptosystem with some redundancies in the public generator matrix. By this view, one can remove the redundancies and recover an alternative secret key in polynomial time by using Overbeck's attack.

Furthermore, since the attack in [HMR15] only considers column scrambler matrices on the base field, one may try to avoid it by combining the reparations proposed in [Loi10, RGH10] with those of [Gab08, GRH09, RGH11]. Nevertheless, our results show that the security of [Gab08, GRH09, RGH11] can be reduced to the one with a column scrambler with entries in the base field. Consequently, using our results and then applying the general attack of [HMR15] may break this "patched" scheme.

All these results put together permit to conclude that all the variants of the GPT scheme based on Gabidulin codes do not represent a secure cryptographic solution.

Chapter 6

$q-$Polynomial Reconstruction Based Cryptosystem

Introduction

In 2005 Faure and Loidreau designed a rank-metric encryption scheme which was not in the McEliece setting. The scheme is very efficient, with small public keys of size a few kiloBytes and with security closely related to the $q-$polynomial reconstruction problem which corresponds to the decoding problem of Gabidulin codes.

We show in this chapter that the Faure-Loidreau scheme is vulnerable to a structural polynomial-time attack that recovers the private key from the public key. Based in part on the security analysis given in [Loi07, Chap. 7], we show that by applying Overbeck's attack on an appropriate public code an attacker can recover the private key very efficiently, only assuming a mild condition on the code, which was always true in all our experimentations.

Informally, the Faure-Loidreau encryption scheme considers three finite fields $\mathbb{F}_q \subset \mathbb{F}_{q^m} \subset \mathbb{L}$. The rank weight of vectors is computed over the field $\mathbb{F}_q$. The public key is then composed of a Gabidulin code of dimension k of length n defined by a matrix $\boldsymbol{G} = (g_{i,j})$ with $g_{i,j} \in \mathbb{F}_{q^m}$ and $\boldsymbol{K} = \boldsymbol{x}\boldsymbol{G} + \boldsymbol{z}$ where $\boldsymbol{x}$ is some vector in $\mathbb{L}^k$ and $\boldsymbol{z}$ is a vector of $\mathbb{L}^n$ with (rank) weight $w > \frac{1}{2}(n-k)$. Both vectors $\boldsymbol{x}$ and $\boldsymbol{z}$ have to be kept secret but from attacker's point of view the private key is *essentially* $\boldsymbol{x}$ since $\boldsymbol{z}$ can be deduced from it.

Our attack uses the Frobenius operator, introduced by Overbeck, which takes as input any vector space $U \subseteq \mathbb{F}_{q^m}^n$ and integer $i \geqslant 1$ in order to construct the vector space $\Lambda_i(U)$ defined as

$$\Lambda_i(U) = U + U^q + \cdots + U^{q^i}.$$

The first step of the attack considers a basis $\gamma_1, \ldots, \gamma_u$ of $\mathbb{L}$ viewed as a vector space over $\mathbb{F}_{q^m}$ of dimension $u > 1$ and defines the vectors $\boldsymbol{v}_i = \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{z})$. Our main result shows that the system can be broken in polynomial time and can be stated as follows:

Theorem 6.1. *If the $\mathbb{F}_{q^m}$-vector space generated by $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_u$ denoted by V satisfies the property*

$$\dim \Lambda_{n-w-k-1}(V) = w \tag{6.1}$$

then the private key $(\boldsymbol{x}, \boldsymbol{z})$ can be recovered from $(\boldsymbol{G}, \boldsymbol{K})$ with $O(n^3)$ operations in the field $\mathbb{L}$.

Notice that if V behaves as random code then generally the condition (6.1) holds. We implemented our attack on parameters given in [FL05, Loi07] for 80-bit security, which were broken in a few seconds. A necessary condition for (6.1) to be true is to choose $u(n-w-k) \geqslant w$ that is to say

$$w \leqslant \frac{u}{u+1}(n-k).$$

This was always the case for parameters proposed in [FL05, Loi07].

Related work. The attack presented in this chapter is very similar to the approach proposed in [LO06] where the authors seek to decode several noisy codewords of a Gabidulin code. Let us assume that we received ℓ words $\boldsymbol{z}_1, \ldots, \boldsymbol{z}_\ell$ from $\mathbb{F}_{q^m}^n$ where each $\boldsymbol{z}_i$ is written as $z_i = \boldsymbol{c}_i + \boldsymbol{e}_i$ with $\boldsymbol{c}_i$ belonging to a Gabidulin code $\mathscr{G}$ of dimension k and length n over $\mathbb{F}_{q^m}$ and the $\boldsymbol{e}_i$'s are vectors from $\mathbb{F}_{q^m}^n$. Let us denote by $\boldsymbol{E}$ the matrix of size $\ell \times n$ formed by the $\boldsymbol{e}_i$'s and let $|\boldsymbol{E}|$ be the dimension of the $\mathbb{F}_q$-vector space generated by the columns of $\boldsymbol{E}$. The authors show that when $|\boldsymbol{E}| \leqslant \frac{\ell}{\ell+1}(n-k)$ then Overbeck's technique recovers in $O(n^3)$ operations the codewords $\boldsymbol{c}_1, \ldots, \boldsymbol{c}_\ell$. It therefore provides a method that decodes a Gabidulin code beyond the classical error-correcting limit $\frac{1}{2}(n-k)$. This approach can be used here to attack the Faure-Loidreau scheme [FL05] because the vectors $\mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_1 K), \ldots, \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_u K)$ can be written as $\boldsymbol{c}_1 + \boldsymbol{v}_1, \ldots, \boldsymbol{c}_u + \boldsymbol{v}_u$ where each $\boldsymbol{c}_i = \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{x})\boldsymbol{G}$ belongs to the Gabidulin code generated by $\boldsymbol{G}$ and the $u \times n$ matrix $\boldsymbol{V}$ formed by $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_u$ satisfies $|\boldsymbol{V}| = w$ which in turn has to verify $w \leqslant \frac{u}{u+1}(n-k)$.

Organisation. In Section 6.1 notations and important notions useful for the chapter are given. In Section 6.2 we present the Faure-Loidreau scheme and in Section 6.3 we describe in full details our attack against it.

6.1 Preliminary Facts

We recall that the field $\mathbb{F}_{q^m}$ can be consider as an $\mathbb{F}_q$-vector space of dimension m. The *trace operator* of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ is the $\mathbb{F}_q$-linear map $\mathbf{Tr}_{\mathbb{F}_{q^m}/\mathbb{F}_q} : \mathbb{F}_{q^m} \longrightarrow \mathbb{F}_q$ defined for any x in $\mathbb{F}_{q^m}$ by

$$\mathbf{Tr}_{\mathbb{F}_{q^m}/\mathbb{F}_q}(x) = x + x^q + \cdots + x^{q^{m-1}}.$$

Let $\mathfrak{B} = \{b_1, \ldots, b_m\}$ be a basis of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$. The *dual* basis, or also called the *trace orthogonal* basis of $\mathfrak{B}$ is a basis $\mathfrak{B}^* = \{b_1^*, \ldots, b_m^*\}$ of $\mathbb{F}_{q^m}$ over $\mathbb{F}_q$ such that for any i and j in $\{1 \ldots, m\}$

$$\mathbf{Tr}_{\mathbb{F}_{q^m}/\mathbb{F}_q}(b_i b_j^*) = \delta_{i,j}$$

where $\delta_{i,i} = 1$ and $\delta_{i,j} = 0$ when $i \neq j$. Note that there always exits a dual basis and furthermore it is possible to express any α from $\mathbb{F}_{q^m}$ as

$$\alpha = \sum_{i=1}^{m} \mathbf{Tr}_{\mathbb{F}_{q^m}/\mathbb{F}_q}(\alpha b_i^*) b_i. \tag{6.2}$$

Definition 6.1 (*Linearized polynomial*)**.** Any univariate polynomial $f \in \mathbb{F}_{q^m}[X]$ of the form

$$f(X) = f_0 X + f_1 X^q + \cdots + f_d X^{q^d}, \quad f_d \neq 0$$

where $0 \leqslant d < m$ is called a $q-$*linearized* polynomial (or $q-$*polynomial*) and d is its $q-$*degree* denoted by $\deg_q(f)$.

The set of $q-$linearized polynomials $f \in \mathbb{F}_{q^m}[X]$ such that $\deg_q(f) < k$ is denoted by $\mathcal{L}_{q,m}^{<k}[X]$.

Definition 6.2 (*Kernel*)**.** The *kernel* $\ker(f)$ of a $q-$*polynomial* f is given by

$$\ker(f) = \{X \in \mathbb{F}_{q^m} \ : \ f(X) = 0\}$$

Theorem 6.2. *The kernel of a $q-$polynomial $f \in \mathbb{F}_{q^m}[X]$ is a $\mathbb{F}_q-$vector subspace of $\mathbb{F}_{q^m}$ with dimension $\dim(\ker(f)) = \deg_q(f)$. Conversely, any $\mathbb{F}_q-$vector subspace $\mathscr{V} \subset \mathbb{F}_{q^m}$ is the kernel of a unique monic $q-$polynomial $f_{\mathscr{V}}$ with $\deg_q(f_{\mathscr{V}}) = \dim \mathscr{V}$.*

A proof (or further references) of this theorem can be found in [Mur14] where it is also proven that for a given $q-$polynomial f, a basis of the kernel $\ker(f)$ can be computed in polynomial time. The converse situation is also true. For a given basis of a $\mathbb{F}_q-$vector subspace $\mathscr{V} \subset \mathbb{F}_{q^m}$, the unique monic $q-$polynomial $f_{\mathscr{V}}$ with kernel $\ker f_{\mathscr{V}} = \mathscr{V}$ can be computed in polynomial time. We understand from this theorem that finding the support $\mathscr{V}$ of an error $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ (in rank metric) is equivalent to find the associated monic $q-$polynomial $f_{\mathscr{V}}$.

In the sequel, any map $h : U \to V$ is naturally extended to vectors $\boldsymbol{x} \in U^n$ by $h(\boldsymbol{x}) = (h(\boldsymbol{x}_1), \ldots, h(\boldsymbol{x}_n))$. This applies in particular to the cases where h is a polynomial or is the Frobenius (and trace) operator. For any subsets $U \subset \mathbb{F}_{q^m}^n$ and $V \subset \mathbb{F}_{q^m}^n$ the notation $U + V$ represents the set

$$U + V = \{\boldsymbol{u} + \boldsymbol{v} \mid \boldsymbol{u} \in U \text{ and } \boldsymbol{v} \in V\}.$$

For any sub-field $\mathbb{K} \subseteq \mathbb{F}_{q^m}$ and $\boldsymbol{x}$ from $\mathbb{F}_{q^m}^n$ the $\mathbb{K}$-vector space generated by $\boldsymbol{x}$ is denoted by $\mathbb{K}\boldsymbol{x}$. For any $\boldsymbol{P} \in \mathsf{GL}_n(\mathbb{K})$ the notation $U\boldsymbol{P}$ is used to denote the set

$$U\boldsymbol{P} = \{\boldsymbol{u}\boldsymbol{P} \mid \boldsymbol{u} \in U\}.$$

For any integer $i \geqslant 0$ we define V^{q^i} as the set of vectors

$$\boldsymbol{v}^{q^i} = (v_1^{q^i}, \ldots, v_n^{q^i})$$

where $\boldsymbol{v}$ describes V. Note that when V is a vector space then V^{q^i} is also a linear subspace of $\mathbb{F}_{q^m}^n$.

Remark 6.1 (Gabidulin codes). Let $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ such that $|\boldsymbol{g}| = n$. The $(n,k)-$Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$ can be defined using linearized polynomials by:

$$\mathscr{G}_k(\boldsymbol{g}) \stackrel{\text{def}}{=} \left\{ \big(f(g_1), ..., f(g_n)\big) \ : \ f \in \mathcal{L}_{q,m}^{<k}[X] \right\}$$

$\boldsymbol{g}$ is called generator vector of $\mathscr{G}_k(\boldsymbol{g})$.

By Remark 6.1 and Theorem 6.2, the decoding problem for Gabidulin codes can be easily translated in terms of $q-$polynomials. Informally, assume that one has to decode a noisy codeword $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$ of a Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$. The problem is (given $\boldsymbol{g}$ and $\boldsymbol{y}$) to find f and $\boldsymbol{e}$ such that $\boldsymbol{y} = f(\boldsymbol{g}) + \boldsymbol{e}$. Let $\mathscr{V}$ be the support of $\boldsymbol{e}$ and L the monic polynomial such that $\ker(L) = \mathscr{V}$. We have

$$L(\boldsymbol{y} - f(\boldsymbol{g})) = 0 \tag{6.3}$$

Thus, to decode $\boldsymbol{y}$, it suffices to find L and f that satisfy equation (6.3). This last problem is called the $q-$polynomial reconstruction problem and can be formally described as follows:

Problem 6.3 ($q-$Polynomial reconstruction)**.** *Given $\boldsymbol{y}$ and $\boldsymbol{g}$ in $\mathbb{F}_{q^m}^n$ together with two integers k and w, the problem is to find a non-zero $q-$polynomial L with $q-$degree at most w and a $q-$polynomial f with $q-$degree at most k such that*

$$L\left[f(\boldsymbol{g}) - \boldsymbol{y}\right] = \boldsymbol{0}$$

In [FL05], Faure and Loidreau proposed a rank-metric encryption scheme based on this problem. The idea is to choose w greater than the error correction capacity of the Gabidulin code $\mathscr{G}_k(\boldsymbol{g})$, in order to avoid the easy instances of problem 6.3 which can be solved by using a decoding algorithm of Gabidulin codes. We describe the scheme in the following section.

6.2 Faure-Loidreau Encryption Scheme

We refer the reader to [FL05] for more details about the system. In fact, the original scheme was described using linearized polynomials. But using the links between linearized polynomials and Gabidulin codes, the system can be described as follows:

Key generation. Throughout this step, besides the fields $\mathbb{F}_q$ and $\mathbb{F}_{q^m}$, another field $\mathbb{L}$ is considered where $\mathbb{L}$ is the extension of $\mathbb{F}_{q^m}$ of degree $u > 1$, and three integers k, n and w such that $u < k < n$ and

$$n - k > w > \left\lfloor \frac{n-k}{2} \right\rfloor. \tag{6.4}$$

1. Pick at random $\boldsymbol{g} \in \mathbb{F}_{q^m}^n$ with $|\boldsymbol{g}| = n$ and let $\boldsymbol{G} \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$ be the generator matrix of $\mathscr{G}_k(\boldsymbol{g}) \subset \mathbb{F}_{q^m}^n$ as in (4.10)
2. Pick at random $\boldsymbol{x} \in \mathbb{L}^k$ such that $\{x_{k-u+1}, \ldots, x_k\}$ form a basis of $\mathbb{L}$ over $\mathbb{F}_{q^m}$
3. Generate randomly $\boldsymbol{s} \in \mathbb{L}^w$ with $|\boldsymbol{s}| = w$ and $\boldsymbol{P} \in \mathsf{GL}_n(\mathbb{F}_q)$ and then compute $\boldsymbol{z} \in \mathbb{L}^n$ defined as
$$\boldsymbol{z} = (\boldsymbol{s} \mid \boldsymbol{0})\, \boldsymbol{P}^{-1}. \tag{6.5}$$

The private key is $(\boldsymbol{x}, \boldsymbol{P})$ and the public key is $(\boldsymbol{g}, k, \boldsymbol{K}, t_{\text{pub}})$ where

$$\boldsymbol{K} = \boldsymbol{x}\boldsymbol{G} + \boldsymbol{z} \qquad \text{and} \qquad t_{\text{pub}} = \left\lfloor \frac{n-w-k}{2} \right\rfloor. \tag{6.6}$$

Encryption. A plaintext here is a vector $\boldsymbol{m} = (m_1, \ldots, m_k)$ belonging to $\mathbb{F}_{q^m}^k$ such that $m_i = 0$ when $i \in \{k-u+1, \ldots, k\}$. To encrypt $\boldsymbol{m}$, one randomly generates $\alpha \in \mathbb{L}$ and $\boldsymbol{e} \in \mathbb{F}_{q^m}^n$ such that $|\boldsymbol{e}| \leqslant t_{\text{pub}}$. The ciphertext is the vector $\boldsymbol{c} \in \mathbb{F}_{q^m}^n$ defined by

$$\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G} + \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha \boldsymbol{K}) + \boldsymbol{e}. \tag{6.7}$$

Decryption. The receiver computes first $\boldsymbol{c}\boldsymbol{P}$ that is to say

$$\begin{aligned} \boldsymbol{c}\boldsymbol{P} &= \boldsymbol{m}\boldsymbol{G}\boldsymbol{P} + \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}\left(\alpha \boldsymbol{x}\boldsymbol{G}\boldsymbol{P} + \alpha \boldsymbol{z}\boldsymbol{P}\right) + \boldsymbol{e}\boldsymbol{P} && (6.8)\\ &= \left(\boldsymbol{m} + \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha \boldsymbol{x})\right)\boldsymbol{G}\boldsymbol{P} + \left(\mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha \boldsymbol{s}) \mid \boldsymbol{0}\right) + \boldsymbol{e}\boldsymbol{P} && (6.9) \end{aligned}$$

Let $\boldsymbol{G}'$ be the $k \times (n-w)$ matrix obtained by removing the first w columns of $\boldsymbol{G}\boldsymbol{P}$ and let $\boldsymbol{e}'$ and $\boldsymbol{c}'$ be respectively the restriction of $\boldsymbol{e}\boldsymbol{P}$ and $\boldsymbol{c}\boldsymbol{P}$ to the last $n-w$ coordinates. We then have

$$\boldsymbol{c}' = \left(\boldsymbol{m} + \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha \boldsymbol{x})\right)\boldsymbol{G}' + \boldsymbol{e}'. \tag{6.10}$$

Using the fact that $\boldsymbol{G}'$ generates a Gabidulin code of length $n-w$ and dimension $k < n-w$ and since $|\boldsymbol{e}'| \leqslant |\boldsymbol{e}| \leqslant \lfloor \frac{1}{2}(n-w-k) \rfloor$, it is possible to recover $\boldsymbol{m}' = \boldsymbol{m} + \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha \boldsymbol{x})$ by applying a decoding algorithm. Since by construction $\boldsymbol{m} \in \mathbb{F}_{q^m}^k$ is chosen so that $m_i = 0$ when $i \in \{k-u+1, \ldots, k\}$ then by choosing a dual basis $\{x_{k-u+1}^*, ..., x_k^*\}$ of $\{x_{k-u+1}, \ldots, x_k\}$ the value of α can be computed as the following

$$\sum_{i=k-u+1}^{k} m_i' x_i^* = \sum_{i=k-u+1}^{k} \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha x_i) x_i^* = \alpha.$$

Once α is recovered, the plaintext $\boldsymbol{m}$ is then equal to $\boldsymbol{m}' - \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\alpha \boldsymbol{x})$.

n	k	u	w	Keys sizes	Transmission rate
56	28	3	16	9408 bits	44%
54	32	4	13	11664 bits	44%

Table 6.1 – Proposed parameters from [Loi07] for the Faure-Loidreau scheme

6.3 Polynomial-Time Key Recovery Attack

In this section, we show that it is possible to recover an alternative private key from the public data $\boldsymbol{K}$ and $\boldsymbol{G}$ when the condition $w \leqslant \frac{u}{u+1}(n-k)$ holds. We start by remarking that if an attacker $\mathbb{A}$ is able to find a matrix $\boldsymbol{T} \in \mathsf{GL}_n(\mathbb{F}_q)$ and $\boldsymbol{z}^* \in \mathbb{L}^w$ such that

$$\boldsymbol{z}\boldsymbol{T} = (\boldsymbol{z}^* \mid \boldsymbol{0}) \text{ and } |\boldsymbol{z}^*| = w$$

then $\mathbb{A}$ can fully recover $\boldsymbol{x} \in \mathbb{L}^k$ by solving only the last $n-w$ equations of the following linear system (see Algorithm 3 for more details)

$$\boldsymbol{K}\boldsymbol{T} = \boldsymbol{x}\boldsymbol{G}\boldsymbol{T} + (\boldsymbol{z}^* \mid \boldsymbol{0}). \tag{6.11}$$

In the sequel, we describe a way to obtain $\boldsymbol{x}$ by finding such a matrix $\boldsymbol{T}$. The first step is to consider a basis $\gamma_1, \ldots, \gamma_u$ of $\mathbb{L}$ viewed as a vector space over $\mathbb{F}_{q^m}$ of dimension $u > 1$. For any $i \in \{1, \ldots, u\}$ we set $\boldsymbol{K}_i = \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{K})$. Lastly, let $\mathscr{C}_{\text{pub}} \subset \mathbb{F}_{q^m}^n$ be the (public) code generated by $\boldsymbol{K}_1, \ldots, \boldsymbol{K}_u$ and $\mathscr{G}_k(\boldsymbol{g})$, that is to say

$$\mathscr{C}_{\text{pub}} = \mathscr{G}_k(\boldsymbol{g}) + \sum_{i=1}^{u} \mathbb{F}_{q^m} \boldsymbol{K}_i. \tag{6.12}$$

Remark 6.2. $\mathscr{C}_{\text{pub}}$ is defined by the generator matrix $\boldsymbol{G}_{\text{pub}}$ where

$$\boldsymbol{G}_{\text{pub}} = \begin{pmatrix} \boldsymbol{G} \\ \boldsymbol{K}_1 \\ \vdots \\ \boldsymbol{K}_u \end{pmatrix} \tag{6.13}$$

For all $i \in \{1, \ldots, u\}$ let us set $\boldsymbol{v}_i = \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{z})$ and $\boldsymbol{b}_i = \left(\mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{s}) \mid \boldsymbol{0}\right) \in \mathbb{F}_{q^m}^n$. By construction, we also have the equality

$$\boldsymbol{v}_i \boldsymbol{P} = \boldsymbol{b}_i. \tag{6.14}$$

Lemma 6.4. *Let us define* $\mathscr{B} = \sum_{i=1}^{m} \mathbb{F}_{q^m} \boldsymbol{b}_i$ *then we have*

$$\mathscr{C}_{\text{pub}} \boldsymbol{P} = \mathscr{G}_k(\boldsymbol{g}\boldsymbol{P}) + \mathscr{B}.$$

Proof. Set $\boldsymbol{x}_i = \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{x}) \in \mathbb{F}_{q^m}^k$. It is sufficient to use Proposition 4.11 and to observe that

$$\begin{aligned} \boldsymbol{K}_i \boldsymbol{P} &= \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{x}) \boldsymbol{G}\boldsymbol{P} + \left(\mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{s}) \mid \mathbf{0}\right) \\ &= \boldsymbol{x}_i \boldsymbol{G}\boldsymbol{P} + \boldsymbol{b}_i \end{aligned}$$

□

Proposition 6.5. *Let $f = n - w - k - 1$ and assume that $\dim \Lambda_f(\mathscr{B}) = w$. The code $\Lambda_f(\mathscr{C}_{\mathrm{pub}})^\perp$ is then of dimension 1 generated by $(\mathbf{0} \mid \boldsymbol{h})\, \boldsymbol{P}^T$ where $\boldsymbol{h} \in \mathbb{F}_{q^m}^{n-w}$ and $|\boldsymbol{h}| = n - w$.*

Furthermore, for any $\widetilde{\boldsymbol{h}} \in \Lambda_f(\mathscr{C}_{\mathrm{pub}})^\perp$ with $\widetilde{\boldsymbol{h}} \neq \mathbf{0}$ and for any $\boldsymbol{T} \in \mathsf{GL}_n(\mathbb{F}_q)$ such that

$$\widetilde{\boldsymbol{h}}(\boldsymbol{T}^{-1})^T = (\mathbf{0} \mid \boldsymbol{h}') \tag{6.15}$$

where $\boldsymbol{h}' \in \mathbb{F}_{q^m}^{n-w}$, there exists $\boldsymbol{z}^* \in \mathbb{F}_{q^m}^w$ with $|\boldsymbol{z}^*| = w$ such that $\boldsymbol{z}\boldsymbol{T} = (\boldsymbol{z}^* \mid \mathbf{0})$.

Proof. Let us decompose $\boldsymbol{G}\boldsymbol{P}$ as $(\boldsymbol{L} \mid \boldsymbol{R})$ where $\boldsymbol{L}$ belongs to $\mathcal{M}_{k,w}(\mathbb{F}_{q^m})$ and $\boldsymbol{R}$ in $\mathcal{M}_{k,n-w}(\mathbb{F}_{q^m})$. Let $\boldsymbol{B} \in \mathcal{M}_{u,w}(\mathbb{F}_{q^m})$ be the matrix where the i-th row is composed by the w first components of $\boldsymbol{b}_i$. Note that $\boldsymbol{G}_{\mathrm{pub}}\boldsymbol{P}$ where $\boldsymbol{G}_{\mathrm{pub}}$ is defined as in (6.13) is a generator matrix of $\mathscr{C}_{\mathrm{pub}}\boldsymbol{P}$, and the following equality holds

$$\boldsymbol{G}_{\mathrm{pub}}\boldsymbol{P} = \begin{pmatrix} \boldsymbol{L} & \boldsymbol{R} \\ \boldsymbol{B} & \mathbf{0} \end{pmatrix}. \tag{6.16}$$

Hence $\Lambda_f(\boldsymbol{G}_{\mathrm{pub}}\boldsymbol{P}) = \Lambda_f(\boldsymbol{G}_{\mathrm{pub}})\boldsymbol{P}$ is a generator matrix of the code $\Lambda_f(\mathscr{C}_{\mathrm{pub}}\boldsymbol{P}) = \Lambda_f(\mathscr{C}_{\mathrm{pub}})\boldsymbol{P}$ which satisfies the equality

$$\Lambda_f(\boldsymbol{G}_{\mathrm{pub}})\boldsymbol{P} = \begin{pmatrix} \Lambda_f(\boldsymbol{L}) & \Lambda_f(\boldsymbol{R}) \\ \Lambda_f(\boldsymbol{B}) & \mathbf{0} \end{pmatrix}.$$

The fact that $\boldsymbol{R}$ generates an $(n-w, k)-$Gabidulin code implies that

$$\mathtt{rank}\left(\Lambda_f(\boldsymbol{R})\right) = k + f = n - w - 1.$$

Consequently, there exists $\boldsymbol{h} \in \mathbb{F}_{q^m}^{n-w}$ with $|\boldsymbol{h}| = n - w$ that satisfies $\Lambda_f\left(\boldsymbol{R}\right)\boldsymbol{h}^T = \mathbf{0}$. Furthermore, the equality $\dim \Lambda_f(\mathscr{B}) = \Lambda_f(\boldsymbol{B})$ holds and implies that

$$\dim \Lambda_f(\mathscr{C}_{\mathrm{pub}})\boldsymbol{P} = \mathtt{rank}\left(\Lambda_f(\boldsymbol{B})\right) + \mathtt{rank}\left(\Lambda_f(\boldsymbol{R})\right) = k + f + w = n - 1.$$

This means that $(\mathbf{0} \mid \boldsymbol{h})$ generates actually the full space $(\Lambda_f(\mathscr{C}_{\mathrm{pub}})\boldsymbol{P})^\perp$ which is equivalent to say $(\mathbf{0} \mid \boldsymbol{h})\, \boldsymbol{P}^T$ generates $\Lambda_f(\mathscr{C}_{\mathrm{pub}})^\perp$.

For the second part of the proposition, let $\widetilde{\boldsymbol{h}}$ be any element from $\Lambda_f(\mathscr{C}_{\mathrm{pub}})^\perp$ with $\widetilde{\boldsymbol{h}} \neq \mathbf{0}$ and let $\boldsymbol{T}$ be in $\mathsf{GL}_n(\mathbb{F}_q)$ such that (6.15) holds for some $\boldsymbol{h}'$ in $\mathbb{F}_q^{n-w}$. There exists an element α in $\mathbb{F}_{q^m}$ such that $\widetilde{\boldsymbol{h}} = (\mathbf{0} \mid \alpha\boldsymbol{h})\boldsymbol{P}^T$. Consider matrices $\boldsymbol{A}_1, \boldsymbol{A}_2$, $\boldsymbol{A}_3$ and $\boldsymbol{A}_4$ such that $\boldsymbol{A}_1 \in \mathcal{M}_{w,w}(\mathbb{F}_q)$ and $\boldsymbol{A}_4 \in \mathcal{M}_{(n-w),(n-w)}(\mathbb{F}_q)$ so that we have

$$\boldsymbol{T}^{-1}\boldsymbol{P} = \begin{pmatrix} \boldsymbol{A}_1 & \boldsymbol{A}_2 \\ \boldsymbol{A}_3 & \boldsymbol{A}_4 \end{pmatrix}.$$

We have then the following equalities

$$(\mathbf{0} \mid \boldsymbol{h}') = \widetilde{\boldsymbol{h}}(\boldsymbol{T}^{-1})^T = (\mathbf{0} \mid \alpha\boldsymbol{h})\,\boldsymbol{P}^T(\boldsymbol{T}^{-1})^T = (\mathbf{0} \mid \alpha\boldsymbol{h})\left(\boldsymbol{T}^{-1}\boldsymbol{P}\right)^T \tag{6.17}$$

It follows from (6.17) that $\boldsymbol{h}\boldsymbol{A}_2^T = \mathbf{0}$ and hence $\boldsymbol{A}_2 = \mathbf{0}$ since $|\boldsymbol{h}| = n - w$. So we can write

$$\boldsymbol{T}^{-1}\boldsymbol{P} = \begin{pmatrix} \boldsymbol{A}_1 & \mathbf{0} \\ \boldsymbol{A}_3 & \boldsymbol{A}_4 \end{pmatrix}.$$

We deduce that $\boldsymbol{P}^{-1}\boldsymbol{T} = \begin{pmatrix} \boldsymbol{A}_1^{-1} & \mathbf{0} \\ -\boldsymbol{A}_4^{-1}\boldsymbol{A}_3\boldsymbol{A}_1^{-1} & \boldsymbol{A}_4^{-1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}' & \mathbf{0} \\ \boldsymbol{C}' & \boldsymbol{D}' \end{pmatrix}$ and consequently, we get

$$\boldsymbol{z}\boldsymbol{T} = (\boldsymbol{s} \mid \mathbf{0})\,\boldsymbol{P}^{-1}\boldsymbol{T} = (\boldsymbol{s} \mid \mathbf{0})\begin{pmatrix} \boldsymbol{A}' & \mathbf{0} \\ \boldsymbol{C}' & \boldsymbol{D}' \end{pmatrix} = (\boldsymbol{s}\boldsymbol{A}' \mid \mathbf{0}).$$

So by letting $\boldsymbol{z}^* = \boldsymbol{s}\boldsymbol{A}' = \boldsymbol{s}\boldsymbol{A}_1^{-1}$ we have proved the proposition. □

Proposition 6.5 shows that an equivalent key can be found in polynomial time by simply using a non zero element of $\Lambda_f(\mathscr{C}_{\text{pub}})^\perp$. We now prove our main result stated in the introduction which shows the weakness of the system.

Theorem 6.6. *If the $\mathbb{F}_{q^m}$-vector space generated by $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_u$ denoted by V satisfies the property*

$$\dim \Lambda_{n-w-k-1}(V) = w$$

then the private key $(\boldsymbol{x}, \boldsymbol{z})$ can be recovered from $(\boldsymbol{G}, \boldsymbol{K})$ with $O(n^3)$ operations in the field $\mathbb{L}$

Proof. Firstly, note that from (6.14) we know that $V\boldsymbol{P} = \mathscr{B}$. Algorithm 3 gives the full description of the attack and provides a proof of Theorem 6.1. Indeed, the attack consists in picking any codeword $\widetilde{\boldsymbol{h}}$ from $\Lambda_{n-w-k-1}(\mathscr{C}_{\text{pub}})^\perp$ and then, by Gaussian elimination, we transform $\widetilde{\boldsymbol{h}}$ so that there exists $\boldsymbol{T} \in \mathsf{GL}_n(\mathbb{F}_q)$ for which we have

$$\widetilde{\boldsymbol{h}}(\boldsymbol{T}^{-1})^T = (\mathbf{0} \mid \boldsymbol{h}')$$

where $\boldsymbol{h}' \in \mathbb{F}_{q^m}^{n-w}$. From Proposition 6.5 we know that $\boldsymbol{T}$ is an equivalent key that will gives an equality of the form (6.11), and therefore it is possible by solving a linear system to find $\boldsymbol{x}$. Lastly, the time complexity comes from the fact the operations involved are essentially Gaussian eliminations over square matrices with n columns and entries in $\mathbb{L}$. □

An important assumption for the success of the attack is that the dimension of $\Lambda_{n-w-k-1}(\mathscr{C}_{\text{pub}})^\perp$ is 1, which was always true in all our experimentations. This assumption is true if and only if the equality $\dim \Lambda_{n-w-k-1}(\mathscr{B}) = w$ holds, which implies to have $u(n-w-k) \geqslant w$, or equivalently

$$w \leqslant \frac{u}{u+1}(n-k). \tag{6.18}$$

Assuming that $\mathscr{B}$ behaves as a random code then $\dim \Lambda_{n-w-k-1}(\mathscr{B}) = w$ would hold with high probability as long as (6.18) is true. The parameters proposed in [Loi07] satisfy (6.18). Furthermore, the analysis given in [Loi07] implies to take $u \geqslant 3$. We implemented the attack with Magma V2.21-6 and the secret key $\boldsymbol{x}$ was found in less than 1 second confirming the efficiency of the approach.

Remark 6.3. Let us observe that taking $w > \frac{u}{u+1}(n-k)$ implies for t_{pub} to be very small since we have

$$t_{\text{pub}} \leqslant \frac{1}{2}(n-w-k) < \frac{1}{2}\left(\frac{n-k}{u+1}\right). \tag{6.19}$$

For instance, with parameters proposed in [Loi07] we would have $t_{\text{pub}} \leqslant 3$. Consequently the values of n, k and m have to be changed so that general decoding attacks fail [GRS16]. Let us notice that this situation is quite similar to the counter-measures proposed in [RGH10, Loi10] to resist to Overbeck's attack. But the strength of this reparation deserves a thorough analysis.

Algorithm 3 Key recovery of Faure-Loidreau scheme where the public key is $(\boldsymbol{G}, \boldsymbol{K})$

$\{\gamma_1, \ldots, \gamma_u\} \leftarrow$ arbitrary basis of $\mathbb{L}$ viewed as a linear space over $\mathbb{F}_{q^m}$
for all $1 \leqslant i \leqslant u$ **do**
  $K_i \leftarrow \mathbf{Tr}_{\mathbb{L}/\mathbb{F}_{q^m}}(\gamma_i \boldsymbol{K})$
end for
Let $\mathscr{C}_{\text{pub}} \subset \mathbb{F}_{q^m}^n$ be the code generated by $\boldsymbol{G}_{\text{pub}}$ ▷ $\boldsymbol{G}_{\text{pub}}$ is defined as in (6.13)
if $\dim \Lambda_{n-w-k-1}(\mathscr{C}_{\text{pub}})^{\perp} = 1$ **then**
  Pick at random $\widetilde{\boldsymbol{h}} \in \Lambda_{n-w-k-1}(\mathscr{C}_{\text{pub}})^{\perp}$
  Compute $\boldsymbol{T} \in \mathsf{GL}_n(\mathbb{F}_q)$ and $\boldsymbol{h}' \in \mathbb{F}_{q^m}^{n-w}$ such that

$$\widetilde{\boldsymbol{h}}(\boldsymbol{T}^{-1})^T = (\boldsymbol{0} \mid \boldsymbol{h}')$$

  $\boldsymbol{K}^* \leftarrow \boldsymbol{K}\boldsymbol{T}$ ▷ $\boldsymbol{K}^* = (\boldsymbol{K}_1^*, \ldots, \boldsymbol{K}_n^*) \in \mathbb{L}^n$
  $\boldsymbol{G}^* \leftarrow \boldsymbol{G}\boldsymbol{T}$ ▷ $\boldsymbol{G}^* = (g_{i,j}^*) \in \mathcal{M}_{k,n}(\mathbb{F}_{q^m})$
  Solve the linear system where $(X_1, \ldots, X_k)$ are the unknowns

$$(\mathcal{L}) : \begin{cases} \boldsymbol{K}_{w+1}^* &= g_{1,w+1}^* X_1 + \cdots + g_{k,w+1}^* X_k \\ &\vdots \\ \boldsymbol{K}_n^* &= g_{1,n}^* X_1 + \cdots + g_{k,n}^* X_k \end{cases}$$

  $\boldsymbol{z} \leftarrow \boldsymbol{K} - \boldsymbol{x}\boldsymbol{G}$ where $\boldsymbol{x}$ is the *unique* solution of $(\mathcal{L})$
end if
return $(\boldsymbol{x}, \boldsymbol{z})$

Table 6.2 – Bound on w with parameters taken from [Loi07] ($m = n$).

n	k	u	w	$\frac{u}{u+1}(n-k)$
56	28	3	16	21
54	32	4	13	17

Conclusion

Faure and Loidreau proposed a rank-metric encryption scheme based on Gabidulin codes related to the problem of the linearized polynomial reconstruction. We showed that the scheme is vulnerable to a polynomial-time key recovery attack by using Overbeck's techniques applied on an appropriate public code.

Our attack assumes that parameters are chosen so that $w \leqslant \frac{u}{u+1}(n-k)$ which was always the case in [FL05, Loi07]. We have also seen that taking $w > \frac{u}{u+1}(n-k)$ implies to choose $t_{\text{pub}} < \frac{1}{2}\left(\frac{n-k}{u+1}\right)$ which exposes further the system to general decoding attacks like [GRS16]. Hence it imposes to increase the key sizes and consequently reduces the practicability of the scheme while offering no assurance that the scheme is still secure. The best choice from a designer's point of view would be to take u as small as possible but a thorough analysis has to be undertaken in light of the connections with the reparations proposed in [RGH10, Loi10]. This point is left as an open question in this chapter and breaking this kind of parameters would lead arguably to a cryptanalysis of [RGH10, Loi10], and to an algorithm that decodes Gabidulin codes beyond the bound $\frac{u}{u+1}(n-k)$.

Chapter 7

Conclusions and Perspectives

7.1 Conclusion

In this thesis, we have studied the security of several code based encryption scheme and mainly McEliece variants. The general idea of the McEliece cryptosystem and its variants is to choose an appropriate private code that will be masked into a public one. This technique opens a general security question: "is the public code distinguishable from a random code ?". A positive answer to that question generally leads to successful structural attacks. That is how several variants of the McEliece based on algebraic codes were proven to be not secured.

In Hamming metric context, one of the most powerful distinguisher is the *square code* (with the component-wise product). This tool has been use to distinguish the public code of several variants of the McEliece encryption schemes. One more, we used this tool in Chapter 3 to distinguish the public code of the modified Sidelnikov cryptosystem [GM13] from a random one, which proved that the system is insecure.

We emphasize that the situation is quiet the same in rank based cryptography where the usual and powerful distinguisher is the operator Λ_i which applies i times the Frobenius operation on the public generator matrix. We have also used this distinguisher in chapters 5 and 6 of this thesis to show that all existing schemes based on Gabidulin codes [Gab08, GRH09, RGH10, RGH11, Loi07, FL05] are actually insecure. However, besides the Gabidulin codes and inspired by the class of MDPC/LDPC codes in Hamming metric, a new class of rank metric codes was recently proposed in [GMRZ13] namely Low Rank Parity Check codes. They are the adaptation of the MDPC/LDPC codes in the rank metric. The LRPC cryptosystem [GMRZ13] is thus the analogue of the MDPC McEliece scheme. The main advantage of the scheme is that it comes, as the MDPC PKC, with a quasi-cyclic version, which allows to drastically reduce the key size. The LRPC scheme is therefore one of the most promising rank-based encryption scheme since it has many security arguments in its favour: compared to the Gabidulin codes, the LRPC codes have a weak algebraic structure and thus seem much more fitted for a cryptographic purpose. Secondly the DC-LRPC scheme is equivalent to the NTRU [HPS98] and thus benefit of a quite long research experience from a cryptanalytic point of view. But the

family of LRPC codes came with a probabilistic decoding algorithm. Furthermore, like in Hamming metric, there is no formal proof of the indistinguishability of the public code from a random one.

7.2 Perspectives

7.2.1 Cryptanalysis

There exist several connections between Generalized Reed-Solomon codes and Gabidulin codes. A natural one is that both GRS and Gabidulin codes are distinguishable from a random code as mentioned at the beginning of this thesis. When looking closely the distinguishers, one can remark that the distinguisher for Gabidulin codes is more general than the distinquisher used for GRS codes. A study of the connections between the two distinguishers, namely the "component-wise product of codes" and the operator Λ_i is to our opinion a promising research perspective since it could allow to find a general distinguisher for Goppa codes. We emphasise that Goppa codes are only distinguishable for some particular parameters such as *high rate Goppa codes* [FOPT10].

7.2.2 Designing

In terms of masking technique, one can remark that the properties of rank-metric allow to see each codeword as a matrix. Exploiting this view for a masking procedure of rank-metric codes might be interesting.

Another alternative branch of research would be to find a new masking technique for which there is a formal proof of the indistinguishability of the public code from a random one, or simply to find a cryptosystem whose the security is based only on the general decoding problem, since putting away this unanswered question of distinguishability can allow to guarantee the security of the system face to structural attacks. Such an idea was already proposed by Alekhnovich [Ale03, Ale11] who considered an innovative approach based on the difficulty of decoding purely random linear codes. Even though the system proposed by Alekhnovich was not practical, several authors were inspired by his work [DMN12, DV13, KMP14] and the recent progress presented in [ABD+16] show the importance of developing this branch of code based cryptography.

Bibliography

[ABD+16] Carlos Aguilar, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, and Gilles Zémor. Efficient encryption from random quasi-cyclic codes. *arXiv preprint arXiv:1612.05572*, 2016.

[AF03] Daniel Augot and Matthieu Finiasz. A public key encryption scheme based on the polynomial reconstruction problem. In *Advances in Cryptology - EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Comput. Sci.*, pages 229–240. Springer, 2003.

[AFL03] Daniel Augot, Matthieu Finiasz, and Pierre Loidreau. Using the trace operator to repair the polynomial reconstruction based cryptosystem. *Eurocrypt*, 2003.

[Ale03] Michael Alekhnovich. More on average case vs approximation complexity. In *44th Symposium on Foundations of Computer Science (FOCS 2003), 11-14 October 2003, Cambridge, MA, USA, Proceedings*, pages 298–307, 2003.

[Ale11] Michael Alekhnovich. More on average case vs approximation complexity. *Computational Complexity*, 20(4):755–786, 2011.

[Bal14] Marco Baldi. *QC-LDPC Code-Based Cryptography*. Springer Briefs in Electrical and Computer Engineering. Springer, 2014.

[BBC08] Marco Baldi, Marco Bodrato, and Franco Chiaraluce. A new analysis of the McEliece cryptosystem based on QC-LDPC codes. In *Proceedings of the 6th international conference on Security and Cryptography for Networks*, SCN '08, pages 246–262, Berlin, Heidelberg, 2008. Springer-Verlag.

[BBC+16] Marco Baldi, Marco Bianchi, Franco Chiaraluce, Joachim Rosenthal, and Davide Schipani. Enhanced public key security for the McEliece cryptosystem. *Journal of Cryptology*, 29(1):1–27, 2016.

[BC07] Marco Baldi and Franco Chiaraluce. Cryptanalysis of a new instance of McEliece cryptosystem based on QC-LDPC codes. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 2591–2595, Nice, France, June 2007.

[BCD+16] Magali Bardet, Julia Chaulet, Vlad Dragoi, Ayoub Otmani, and Jean-Pierre Tillich. Cryptanalysis of the McEliece public key cryptosystem based on polar codes. In *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings*, pages 118–143, 2016.

[BCGO09] Thierry P. Berger, Pierre-Louis Cayrel, Philippe Gaborit, and Ayoub Otmani. Reducing key length of the McEliece cryptosystem. In Bart Preneel, editor, *Progress in Cryptology - AFRICACRYPT 2009*, volume 5580 of *Lecture Notes in Comput. Sci.*, pages 77–97, Gammarth, Tunisia, June 21-25 2009.

[BDLO16] Magali Bardet, Vlad Dragoi, Jean-Gabriel Luque, and Ayoub Otmani. Weak keys for the quasi-cyclic MDPC public key encryption scheme. In *Progress in Cryptology - AFRICACRYPT 2016 - 8th International Conference on Cryptology in Africa, Fes, Morocco, April 13-15, 2016, Proceedings*, pages 346–367, 2016.

[BGJT14] Razvan Barbulescu, Pierrick Gaudry, Antoine Joux, and Emmanuel Thomé. A heuristic quasi-polynomial algorithm for discrete logarithm in finite fields of small characteristic. In *Advances in Cryptology - EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Comput. Sci.*, pages 1–16, Copenhagen, Denmark, May 2014. Springer.

[BJMM12] Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding. In *Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings*, pages 520–536, 2012.

[BL05] Thierry P. Berger and Pierre Loidreau. How to mask the structure of codes for a cryptographic use. *Designs, Codes and Cryptography*, 35(1):63–79, 2005.

[BLP08] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Attacking and defending the McEliece cryptosystem. In *Post-Quantum Cryptography 2008*, volume 5299 of *Lecture Notes in Comput. Sci.*, pages 31–46, 2008.

[BLP10] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Wild McEliece. In Alex Biryukov, Guang Gong, and DouglasR. Stinson, editors, *Selected Areas in Cryptography*, volume 6544 of *Lecture Notes in Comput. Sci.*, pages 143–158, 2010.

[BLP11] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Wild McEliece Incognito. In Bo-Yin Yang, editor, *Post-Quantum Cryptography 2011*,

volume 7071 of *Lecture Notes in Comput. Sci.*, pages 244–254. Springer Berlin Heidelberg, 2011.

[BML13] Paulo S. L. M. Barreto, Rafael Misoczki, and Richard Lindner. Decoding square-free Goppa codes over $\mathbb{F}_p$. *IEEE Trans. Information Theory*, 59(10):6851–6858, 2013.

[BMvT78] Elwyn Berlekamp, Robert McEliece, and Henk van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Inform. Theory*, 24(3):384–386, May 1978.

[CB13] Ivan V. Chizhov and Mikhail A. Borodin. The failure of McEliece PKC based on Reed-Muller codes. IACR Cryptology ePrint Archive, Report 2013/287, 2013. `http://eprint.iacr.org/`.

[CB14] Ivan V. Chizhov and Mikhail A. Borodin. Effective attack on the McEliece cryptosystem based on Reed-Muller codes. *Discrete Math. Appl.*, 24(5):273–280, 2014.

[CFS01] Nicolas Courtois, Matthieu Finiasz, and Nicolas Sendrier. How to achieve a McEliece-based digital signature scheme. In *Advances in Cryptology - ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Comput. Sci.*, pages 157–174, Gold Coast, Australia, 2001. Springer.

[CGG+14] Alain Couvreur, Philippe Gaborit, Valérie Gauthier-Umaña, Ayoub Otmani, and Jean-Pierre Tillich. Distinguisher-based attacks on public-key cryptosystems using Reed-Solomon codes. *Designs, Codes and Cryptography*, 73(2):641–666, 2014.

[CMCP14] Alain Couvreur, Irene Márquez-Corbella, and Ruud Pellikaan. A polynomial time attack against algebraic geometry code based public key cryptosystems. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT 2014*, pages 1446–1450, June 2014.

[Cor04] Jean-Sébastien Coron. Cryptanalysis of a public-key encryption scheme based on the polynomial reconstruction problem. In *Public Key Cryptography - PKC 2004, 7th International Workshop on Theory and Practice in Public Key Cryptography, Singapore, March 1-4, 2004*, pages 14–27, 2004.

[COT14a] Alain Couvreur, Ayoub Otmani, and Jean-Pierre Tillich. New identities relating wild Goppa codes. *Finite Fields Appl.*, 29:178–197, 2014.

[COT14b] Alain Couvreur, Ayoub Otmani, and Jean-Pierre Tillich. Polynomial time attack on wild McEliece over quadratic extensions. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Comput. Sci.*, pages 17–39. Springer Berlin Heidelberg, 2014.

[CS96] Florent Chabaud and Jacques Stern. The cryptographic security of the syndrome decoding problem for rank distance codes. In *Advances in Cryptology - ASIACRYPT 1996*, volume 1163 of *Lecture Notes in Comput. Sci.*, pages 368–381, Kyongju, Korea, November 1996. Springer.

[CT16] Irene Marquez Corbella and Jean-Pierre Tillich. Using Reed-Solomon codes in the (U | U + V) construction and an application to cryptography. In *IEEE International Symposium on Information Theory, ISIT 2016, Barcelona, Spain, July 10-15, 2016*, pages 930–934, 2016.

[DH76] Whitfield Diffie and Martin Hellman. New directions in cryptography. *IEEE Trans. Inform. Theory*, 22(6):644–654, November 1976.

[DMN12] Nico Döttling, Jörn Müller-Quade, and Anderson C. A. Nascimento. IND-CCA secure cryptography based on a variant of the LPN problem. In *Advances in Cryptology - ASIACRYPT 2012*, volume 7658 of *Lecture Notes in Comput. Sci.*, pages 485–503, Beijing, China, 2012. Springer.

[DV13] Alexandre Duc and Serge Vaudenay. HELEN: A public-key cryptosystem based on the LPN and the decisional minimal distance problems. In *Progress in Cryptology - AFRICACRYPT 2013*, volume 7918 of *Lecture Notes in Comput. Sci.*, pages 107–126. Springer, 2013.

[FGO+13] Jean-Charles Faugère, Valérie Gauthier, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. A distinguisher for high rate McEliece cryptosystems. *IEEE Trans. Inform. Theory*, 59(10):6830–6844, October 2013.

[FL05] Cédric Faure and Pierre Loidreau. A new public-key cryptosystem based on the problem of reconstructing p-polynomials. In *Coding and Cryptography, International Workshop, WCC 2005, Bergen, Norway, March 14-18, 2005. Revised Selected Papers*, pages 304–315, 2005.

[FLdP08] Jean-Charles Faugère, Françoise Levy-dit-Vehel, and Ludovic Perret. Cryptanalysis of Minrank. In David Wagner, editor, *Advances in Cryptology - CRYPTO 2008*, volume 5157 of *Lecture Notes in Comput. Sci.*, pages 280–296, 2008.

[FM08] Cédric Faure and Lorenz Minder. Cryptanalysis of the McEliece cryptosystem over hyperelliptic curves. In *Proceedings of the eleventh International Workshop on Algebraic and Combinatorial Coding Theory*, pages 99–107, Pamporovo, Bulgaria, June 2008.

[FOP+14] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, Frédéric de Portzamparc, and Jean-Pierre Tillich. Structural weakness of compact variants of the McEliece cryptosystem. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT 2014*, pages 1717–1721, Honolulu, HI, USA, July 2014.

[FOP+16a] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, Frédéric de Portzamparc, and Jean-Pierre Tillich. Folding alternant and Goppa Codes with non-trivial automorphism groups. *IEEE Trans. Inform. Theory*, 62(1):184–198, 2016.

[FOP+16b] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, Frédéric de Portzamparc, and Jean-Pierre Tillich. Structural cryptanalysis of McEliece schemes with compact keys. *Designs, Codes and Cryptography*, 79(1):87–112, 2016.

[FOPT10] Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. Algebraic cryptanalysis of McEliece variants with compact keys. In *Advances in Cryptology - EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Comput. Sci.*, pages 279–298, 2010.

[Gab05] Philippe Gaborit. Shorter keys for code based cryptography. In *Proceedings of the 2005 International Workshop on Coding and Cryptography (WCC 2005)*, pages 81–91, Bergen, Norway, March 2005.

[Gab08] Ernst. M. Gabidulin. Attacks and counter-attacks on the GPT public key cryptosystem. *Designs, Codes and Cryptography*, 48(2):171–177, 2008.

[Gal63] R. G. Gallager. *Low Density Parity Check Codes*. M.I.T. Press, Cambridge, Massachusetts, 1963.

[Gao03] Shuhong Gao. A new algorithm for decoding Reed-Solomon codes. *Communications, Information and Network Security*, 2003.

[Gib95] Keith Gibson. Severely denting the Gabidulin version of the McEliece public key cryptosystem. *Designs, Codes and Cryptography*, 6(1):37–45, 1995.

[Gib96] Keith Gibson. The security of the Gabidulin public key cryptosystem. In Ueli Maurer, editor, *Advances in Cryptology - EUROCRYPT '96*, volume 1070 of *Lecture Notes in Comput. Sci.*, pages 212–223. Springer, 1996.

[GM13] Cheikh Thiecoumba Gueye and El Hadji Modou Mboup. Secure cryptographic scheme based on modified Reed Muller codes. *International Journal of Security and Its Applications*, 7(3):55–64, 2013.

[GMRZ13] Philippe Gaborit, Gaétan Murat, Olivier Ruatta, and Gilles Zémor. Low rank parity check codes and their application to cryptography. In *Proceedings of the Workshop on Coding and Cryptography WCC'2013*, Bergen, Norway, 2013. Available on `www.selmer.uib.no/WCC2013/pdfs/Gaborit.pdf`.

[GO01] Ernst M. Gabidulin and Alexei V. Ourivski. Modified GPT PKC with right scrambler. *Electron. Notes Discrete Math.*, 6:168–177, 2001.

[GOHA03] Ernst M. Gabidulin, Alexei V. Ourivski, Bahram Honary, and Bassem Ammar. Reducible rank codes and their applications to cryptography. *IEEE Trans. Inform. Theory*, 49(12):3289–3293, 2003.

[Gop70] Valerii Denisovich Goppa. A new class of linear correcting codes. *Problemy Peredachi Informatsii*, 6(3):24–30, 1970.

[GOT12a] Valérie Gauthier, Ayoub Otmani, and Jean-Pierre Tillich. A distinguisher-based attack of a homomorphic encryption scheme relying on Reed-Solomon codes. *CoRR*, abs/1203.6686, 2012.

[GOT12b] Valérie Gauthier, Ayoub Otmani, and Jean-Pierre Tillich. A distinguisher-based attack on a variant of McEliece's cryptosystem based on Reed-Solomon codes. *CoRR*, abs/1204.6459, 2012.

[GOTK16] Philippe Gaborit, Ayoub Otmani, and Hervé Talé-Kalachi. Polynomial-time key recovery attack on the faure-loidreau scheme based on gabidulin codes. *CoRR*, abs/1606.07760, 2016.

[GP13] Ernst Gabidulin and Nina Pilipchuk. GPT cryptosystem for information network security. In *International Conference on Information Society (i-Society 2013)*, number 8, pages 21–25, 2013.

[GP14] Ernst Gabidulin and Nina Pilipchuk. Modified GPT cryptosystem for information network security. *International Journal for Information Security Research*, 4(8):937–946, 2014.

[GPT91] Ernst M. Gabidulin, A. V. Paramonov, and O. V. Tretjakov. Ideals over a non-commutative ring and their applications to cryptography. In *Advances in Cryptology - EUROCRYPT'91*, number 547 in Lecture Notes in Comput. Sci., pages 482–489, Brighton, April 1991.

[GRH09] Ernst Gabidulin, Haitam Rashwan, and Bahram Honary. On improving security of GPT cryptosystems. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 1110–1114. IEEE, 2009.

[GRS16] Philippe Gaborit, Olivier Ruatta, and Julien Schrek. On the complexity of the rank syndrome decoding problem. *IEEE Trans. Information Theory*, 62(2):1006–1019, 2016.

[GZ16] Philippe Gaborit and Gilles Zémor. On the hardness of the decoding and the minimum distance problems for rank codes. *IEEE Trans. Information Theory*, 62(12):7245–7252, 2016.

[HMR15] Anna-Lena Horlemann-Trautmann, Kyle Marshall, and Joachim Rosenthal. Extension of overbeck's attack for gabidulin based cryptosystems. *CoRR*, abs/1511.01549, 2015.

[HPS98] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A ring-based public key cryptosystem. In Joe Buhler, editor, *Algorithmic Number Theory, Third International Symposium, ANTS-III, Portland, Oregon, USA, June 21-25, 1998, Proceedings*, volume 1423 of *Lecture Notes in Comput. Sci.*, pages 267–288. Springer, 1998.

[HSEA14] R Hooshmand, M Koochak Shooshtari, T Eghlidos, and MR Aref. Reducing the key length of McEliece cryptosystem using polar codes. In *2014 11th International Conference on Information Security and Cryptology (ISCISC)*, pages 104–108. IEEE, 2014.

[HTMR16] Anna-Lena Horlemann-Trautmann, Kyle Marshall, and Joachim Rosenthal. Considerations for rank-based cryptosystems. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 2544–2548. IEEE, 2016.

[JM96] Heeralal Janwa and Oscar Moreno. McEliece public key cryptosystems using algebraic-geometric codes. *Designs, Codes and Cryptography*, 8(3):293–307, 1996.

[KKM+17] S. Kudekar, S. Kumar, M. Mondelli, H. D. Pfister, E. Sasoglu, and R. Urbanke. Reed-Muller codes achieve capacity on erasure channels. *IEEE Transactions on Information Theory*, PP(99):1–1, 2017.

[KMP14] Eike Kiltz, Daniel Masny, and Krzysztof Pietrzak. Simple chosen-ciphertext security from low-noise LPN. In *International Workshop on Public Key Cryptography*, pages 1–18. Springer, 2014.

[KY04] Aggelos Kiayias and Moti Yung. Cryptanalyzing the polynomial-reconstruction based public-key system under optimal parameter choice. In *Advances in Cryptology - ASIACRYPT 2004, 10th International Conference on the Theory and Application of Cryptology and Information Security, Jeju Island, Korea, December 5-9, 2004, Proceedings*, pages 401–416, 2004.

[LB88] Pil Joong Lee and Ernest F. Brickell. An observation on the security of McEliece's public-key cryptosystem. In *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*, pages 275–280, 1988.

[LdVP06] Françoise Lévy-dit Vehel and Ludovic Perret. Algebraic decoding of codes in rank metric. In *proceedings of YACC06*, Porquerolles, France, June 2006.

[LDW94] Yuan Xing Li, Robert H. Deng, and Xin Mei Wang. On the equivalence of McEliece's and Niederreiter's public-key cryptosystems. *IEEE Trans. Inform. Theory*, 40(1):271–273, 1994.

[Leo88] Jeffrey Leon. A probabilistic algorithm for computing minimum weights of large error-correcting codes. *IEEE Trans. Inform. Theory*, 34(5):1354–1359, 1988.

[LJ12] Carl Löndahl and Thomas Johansson. A new version of McEliece PKC based on convolutional codes. In *Information and Communications Security, ICICS*, volume 7168 of *Lecture Notes in Comput. Sci.*, pages 461–470. Springer, 2012.

[LO06] Pierre Loidreau and Raphael Overbeck. Decoding rank errors beyond the error-correction capability. In *Proceedings of the Tenth International Workshop on Algebraic and Combinatorial Coding Theory, ACCT-10*, pages 168–190, 2006.

[Loi07] Pierre Loidreau. *Rank metric and cryptography.* Accreditation to supervise research, Université Pierre et Marie Curie - Paris VI, January 2007.

[Loi10] Pierre Loidreau. Designing a rank metric based McEliece cryptosystem. In Nicolas Sendrier, editor, *Post-Quantum Cryptography 2010*, volume 6061 of *Lecture Notes in Comput. Sci.*, pages 142–152. Springer, 2010.

[LT13] Grégory Landais and Jean-Pierre Tillich. An efficient attack of a McEliece cryptosystem variant based on convolutional codes. In P. Gaborit, editor, *Post-Quantum Cryptography'13*, volume 7932 of *Lecture Notes in Comput. Sci.*, pages 102–117. Springer, June 2013.

[MB09] Rafael Misoczki and Paulo Barreto. Compact McEliece keys from Goppa codes. In *Selected Areas in Cryptography*, Calgary, Canada, August 13-14 2009.

[McE78] Robert J. McEliece. *A Public-Key System Based on Algebraic Coding Theory*, pages 114–116. Jet Propulsion Lab, 1978. DSN Progress Report 44.

[Min07] Lorenz Minder. *Cryptography based on error correcting codes.* PhD thesis, Ecole Polytechnique Fédérale de Lausanne, 2007.

[Mit51] N Mitani. On the transmission of numbers in a sequential computer. In *National Convention of the Institute of Electrical Communication Engineers of Japan, November*, 1951.

[MMT11] Alexander May, Alexander Meurer, and Enrico Thomae. Decoding random linear codes in $O(2^{0.054n})$. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology - ASIACRYPT 2011*, volume 7073 of *Lecture Notes in Comput. Sci.*, pages 107–124. Springer, 2011.

[MRAS00] Chris Monico, Joachim Rosenthal, and Amin A. Shokrollahi. Using low density parity check codes in the McEliece cryptosystem. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, page 215, Sorrento, Italy, 2000.

[MS86] Florence J. MacWilliams and Neil J. A. Sloane. *The Theory of Error-Correcting Codes*. North–Holland, Amsterdam, fifth edition, 1986.

[MS07] Lorenz Minder and Amin Shokrollahi. Cryptanalysis of the Sidelnikov cryptosystem. In *Advances in Cryptology - EUROCRYPT 2007*, volume 4515 of *Lecture Notes in Comput. Sci.*, pages 347–360, Barcelona, Spain, 2007.

[MTSB13] Rafael Misoczki, Jean-Pierre Tillich, Nicolas Sendrier, and Paulo S. L. M. Barreto. MDPC-McEliece: New McEliece variants from moderate density parity-check codes. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 2069–2073, 2013.

[Mul54] D. E. Muller. Application of boolean algebra to switching circuit design and to error detection. *Transactions of the I.R.E. Professional Group on Electronic Computers*, EC-3(3):6–12, Sept 1954.

[Mur14] Gaétan Murat. *Résultants de polynômes de Ore et Cryptosystèmes de McEliece sur des Codes Rang faiblement structurés*. PhD thesis, Université de Limoges, 2014.

[Nie86] Harald Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problems of Control and Information Theory*, 15(2):159–166, 1986.

[NMBB10] Robert Niebuhr, Mohammed Meziani, Stanislav Bulygin, and Johannes Buchmann. Selecting parameters for secure McEliece-based cryptosystems. IACR Cryptology ePrint Archive, Report2010/271, 2010.

[OB09] Samuel Ouzan and Yair Be'ery. Moderate-density parity-check codes. *arXiv preprint arXiv:0911.3262*, 2009.

[OJ02] Alexei V. Ourivski and Thomas Johansson. New technique for decoding codes in the rank metric and its cryptography applications. *Problems of Information Transmission*, 38(3):237–246, 2002.

[OTD08] Ayoub Otmani, Jean-Pierre Tillich, and Léonard Dallot. Cryptanalysis of McEliece cryptosystem based on quasi-cyclic LDPC codes. In

Proceedings of First International Conference on Symbolic Computation and Cryptography, pages 69–81, Beijing, China, April 28-30 2008. LMIB Beihang University.

[OTK15] Ayoub Otmani and Hervé Talé-Kalachi. Square code attack on a modified sidelnikov cryptosystem. In *Codes, Cryptology, and Information Security - First International Conference, C2SI 2015, Rabat, Morocco, May 26-28, 2015, Proceedings - In Honor of Thierry Berger*, pages 173–183, 2015.

[OTKN16] Ayoub Otmani, Hervé Talé-Kalachi, and Sélestin Ndjeya. Improved cryptanalysis of rank metric schemes based on Gabidulin codes. *CoRR*, abs/1602.08549, 2016.

[Ove05a] Raphael Overbeck. Extending Gibson's attacks on the GPT cryptosystem. In Oyvind Ytrehus, editor, *WCC 2005*, volume 3969 of *Lecture Notes in Comput. Sci.*, pages 178–188. Springer, 2005.

[Ove05b] Raphael Overbeck. A new structural attack for GPT and variants. In *Mycrypt*, volume 3715 of *Lecture Notes in Comput. Sci.*, pages 50–63, 2005.

[Ove08] Raphael Overbeck. Structural attacks for public key cryptosystems based on Gabidulin codes. *J. Cryptology*, 21(2):280–301, 2008.

[Pat75] N. Patterson. The algebraic decoding of Goppa codes. *IEEE Trans. Inform. Theory*, 21(2):203–207, 1975.

[Per12] Edoardo Persichetti. Compact McEliece keys based on quasi-dyadic Srivastava codes. *J. Math. Cryptol.*, 6(2):149–169, 2012.

[Pra62] Eugene Prange. The use of information sets in decoding cyclic codes. *IRE Transactions on Information Theory*, 8(5):5–9, 1962.

[Ree54] I. S. Reed. A class of multiple-error-correcting codes and the decoding scheme. *IRE Trans.*, IT-4:38–49, 1954.

[RGH10] Haitam Rashwann, Ernst Gabidulin, and Bahram Honary. A smart approach for GPT cryptosystem based on rank codes. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 2463–2467. IEEE, 2010.

[RGH11] Haitam Rashwan, Ernst Gabidulin, and Bahram Honary. Security of the GPT cryptosystem and its applications to cryptography. *Security and Communication Networks*, 4(8):937–946, 2011.

[RS60] Irvin Stoy Reed and Gustave Solomon. Polynomial codes over certain finite fields. *Journal of the Society for Industrial and Applied Mathematics*, 8:300–304, 1960.

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

[Sen02] Nicolas Sendrier. Cryptosystèmes à clé publique basés sur les codes correcteurs d'erreurs. In *Mémoire d'habilitation à diriger des recherches, Université Paris 6*, 2002.

[Sho94] P.W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In S. Goldwasser, editor, *FOCS*, pages 124–134, 1994.

[Sho97] Peter W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, 1997.

[Sid94] Vladimir Michilovich Sidelnikov. A public-key cryptosytem based on Reed-Muller codes. *Discrete Math. Appl.*, 4(3):191–207, 1994.

[SK14] Sujan Raj Shrestha and Young-Sik Kim. New McEliece cryptosystem based on polar codes as a candidate for post-quantum cryptography. In *2014 14th International Symposium on Communications and Information Technologies (ISCIT)*, pages 368–372. IEEE, 2014.

[SS92] Vladimir Michilovich Sidelnikov and S.O. Shestakov. On the insecurity of cryptosystems based on generalized Reed-Solomon codes. *Discrete Math. Appl.*, 1(4):439–444, 1992.

[Ste88] Jacques Stern. A method for finding codewords of small weight. In G. D. Cohen and J. Wolfmann, editors, *Coding Theory and Applications*, volume 388 of *Lecture Notes in Comput. Sci.*, pages 106–113. Springer, 1988.

[UG14] Alexey Urivskiy and Ernst Gabidulin. On the equivalence of different variants of the GPT cryptosystem. Number 3, pages 95–97. IEEE, 2014.

[Wan16] Yongge Wang. Quantum resistant random linear code based public key encryption scheme rlce. In *Information Theory (ISIT), 2016 IEEE International Symposium on*, pages 2519–2523. IEEE, 2016.

[Wie06a] Christian Wieschebrink. An attack on a modified Niederreiter encryption scheme. In Moti Yung, Yevgeniy Dodis, Aggelos Kiayias, and Tal Malk, editors, *Public-Key Cryptography - PKC 2006*, volume 3958 of *Lecture Notes in Comput. Sci.*, pages 14–26. Springer, 2006.

[Wie06b] Christian Wieschebrink. Two NP-complete problems in coding theory with an application in code based cryptography. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 1733–1737, 2006.

[Wie09] Christian Wieschebrink. Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. IACR Cryptology ePrint Archive, Report 2009/452, 2009.

[Wie10] Christian Wieschebrink. Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. In *Post-Quantum Cryptography 2010*, volume 6061 of *Lecture Notes in Comput. Sci.*, pages 61–72. Springer, 2010.