Philippe Gaborit and Jean-Christophe Deneuville

Code-based Cryptography

Contributors

Philippe Gaborit
XLIM-MATHIS
University of Limoges, France

Jean-Christophe Deneuville
École Nationale de l'Aviation Civile
Federal University of Toulouse,
France

Chapter 1

Code-based Cryptography

Introduction

Code-based cryptography refers to asymmetric cryptography based on coding theory assumptions, meaning that the underlying primitives can be reduced to solving decoding problems. This chapter focuses on the essentials of code-based cryptography, and covers some reminders on coding theory and the associated hard problems useful for cryptography, historical constructions such as the McEliece [McE78] and Niederreiter [Nie86] constructions, as well as more recent proposals for Public Key Cryptography, Key Exchange protocols, and Digital Signatures.

With the announced downfall of number theoretic based approaches to provide secure public key primitives against quantum adversaries [Sho97, Gro96], the field of post-quantum cryptography has received growing interest with a lot of research efforts. Among the possible alternative tools such as lattices, hash functions, multivariate polynomials or isogenies of elliptic curves, coding theory stands as a credible mature candidate for several reasons that will be discussed in the next sections: it allows to design most of the usual (most used) primitives, the complexity of best known attacks is well-studied and rather stable, and offers time/memory/security trade-offs.

In particular, in the recent years a lot of progress was made, and beside the classical McEliece framework which has been known from 1978 with its advantages and drawbacks, some new approaches introduced the possibilty to have code-based cryptosystems with small key sizes and with security reductions to generic problems a way to avoid the classical strutural attacks associated to the McEliece framework.

At last the recent standardization process launched by the Nist Institute of Standard Technology (NIST) in 2017 [NIS17] has thrown code-based cryptography into a new era by forcing it to consider real applications, which had not really been done before, because of the practical efficiency of number theory based cryptosystems like RSA.

1.1 Preliminaries

1.1.1 Notation

Throughout this chapter, $\mathbb{Z}$ denotes the ring of integers and $\mathbb{F}_q$ denotes the finite field of q elements for q a power of a prime. Let $\mathcal{R} = \mathbb{F}_q[X]/(X^n - 1)$ denote the quotient ring of polynomials modulo $X^n - 1$ whose coefficients lie in $\mathbb{F}_q$. Elements of $\mathcal{R}$ will be interchangeably considered as row vectors or polynomials. Vectors/Polynomials (resp. matrices) will be represented by lower-case (resp. upper-case) bold letters[1].

Typical case: a typical case is $q = 2$ and n a prime number, chosen such that 2 is primitive modulo n, indeed in that case $X^n - 1$ is factorized as the product of two irreducible polynomials, $X^n - 1 = (X+1)(X^{n-1} + X^{n-2} + \cdots + 1)$ [GZ08]. Hence $X^n - 1$ can be considered as 'almost' irreducible, which is considered to be better for security and makes easier the fact that elements of $\mathcal{R}$ are invertible modulo $X^N - 1$, since in that case invertible elements of $\mathcal{R}$ are exactly polynomials with an odd number of binary coefficients.

For any two elements $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{R}$, their product, defined as the usual product of two polynomials modulo $X^n - 1$, is as follows: $\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{z} \in \mathcal{R}$ with

$$z_k = \sum_{i+j \equiv k \mod n} x_i y_j, \text{ for } i, j, k \in \{0, \ldots, n-1\}. \tag{1.1}$$

Notice that as the product of two elements over the *commutative* ring $\mathcal{R}$, we have $\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{y} \cdot \boldsymbol{x}$.

For any finite set $\mathcal{S}$, $x \xleftarrow{\$} \mathcal{S}$ denotes a uniformly random element sampled from $\mathcal{S}$. For any $x \in \mathbb{R}$, let $\lfloor x \rfloor$ denotes the biggest integer smaller than or equal to x. Finally, all logarithms $\log(\cdot)$ will be base-2 unless explicitly mentioned.

Definition 1 (Circulant Matrix) *Let $\boldsymbol{x} = (x_0, \ldots, x_{n-1}) \in \mathbb{F}_q^n$. The* circulant matrix *induced by $\boldsymbol{x}$ is defined and denoted as follows:*

$$\mathbf{rot}(\boldsymbol{x}) = \begin{pmatrix} x_0 & x_1 & \cdots & x_{n-1} \\ x_{n-1} & x_0 & \cdots & x_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ x_1 & x_2 & \cdots & x_0 \end{pmatrix} \in \mathbb{F}_q^{n \times n} \tag{1.2}$$

As a consequence, it is easy to see that the product of any two elements $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{R}$ can be expressed as a usual vector-matrix (or matrix-vector) product using the $\mathbf{rot}(\cdot)$ operator as

$$\boldsymbol{x} \cdot \boldsymbol{y} = \boldsymbol{x} \times \mathbf{rot}(\boldsymbol{y}) = \left(\mathbf{rot}(\boldsymbol{x})^\top \times \boldsymbol{y}^\top\right)^\top = \boldsymbol{y} \times \mathbf{rot}(\boldsymbol{x}) = \boldsymbol{y} \cdot \boldsymbol{x}. \tag{1.3}$$

[1] Vectors are usually considered in row representation in coding theory, against column representation in lattice-based cryptography.

1.1.2 Background on coding theory

We now recall some basic definitions and properties about coding theory, that are regularly used in code-based cryptography. More details about coding theory can be found in dedicated manuscripts such as [GRS12], [HP10], or [MS77].

Definition 2 (Linear code) *A* linear code $\mathcal{C}$ *of length* n *and dimension* k *(denoted* $[n,k]$*) is a subspace of the vector space* $\mathbb{F}_q^n$ *of dimension* k*. Elements of* $\mathcal{C}$ *are referred to as codewords.*

Definition 3 (Scalar product) *Let* $x(x_1,...,x_n)$ *and* $y(y_1,...,y_n)$ *be two elements of* $\mathbb{F}_q^n$ *then the scalar product of* x *and* y *is defined as*

$$\langle \boldsymbol{x}, \boldsymbol{c} \rangle = \sum_{i=0}^{n} x_i.y_i$$

Definition 4 (Dual code) *The* dual $\mathcal{C}^\perp$ *of a linear code* $\mathcal{C}[n,k]$ *is the linear code of length* n *and dimension* $n-k$ *defined by:*

$$\mathcal{C}^\perp = \left\{ \boldsymbol{x} \in \mathbb{F}_q^n \text{ s.t. } \langle \boldsymbol{x}, \boldsymbol{c} \rangle = 0, \forall \boldsymbol{c} \in \mathcal{C} \right\}. \tag{1.4}$$

Definition 5 (Generator Matrix) *We say that* $\boldsymbol{G} \in \mathbb{F}_q^{k \times n}$ *is a* Generator Matrix *for the* $[n,k]$ *code* $\mathcal{C}$ *if*

$$\mathcal{C} = \left\{ \boldsymbol{mG}, \text{ for } \boldsymbol{m} \in \mathbb{F}_q^k \right\}. \tag{1.5}$$

Definition 6 (Parity-Check Matrix) *Given an* $[n,k]$ *code* $\mathcal{C}$*, we say that* $\boldsymbol{H} \in \mathbb{F}_q^{(n-k)\times n}$ *is a* Parity-Check Matrix *for* $\mathcal{C}$ *if* $\boldsymbol{H}$ *is a generator matrix of the dual code* $\mathcal{C}^\perp$*, or more formally, if*

$$\mathcal{C}^\perp = \left\{ \boldsymbol{x} \in \mathbb{F}_q^n \text{ such that } \boldsymbol{Hx}^\top = \boldsymbol{0} \right\}, \tag{1.6}$$

where $\boldsymbol{Hx}^\top$ *is called the* syndrome *of* $\boldsymbol{x}$*.*

Proposition 1 *It follows from the previous definitions that* $\boldsymbol{GH}^\top = \boldsymbol{0}$*.*

Code-based cryptography was originally proposed using the standard Hamming norm. Other norms (such as the rank metric) allow for the design of cryptographic primitives with different properties, we focus on the Hamming metric for most of this chapter and refer the reader to Sec. 1.10 for a brief overview of rank metric.

Definition 7 (Hamming weight and distance) *Let* $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_q^n$ *be two words. The Hamming weight of* $\boldsymbol{x}$ *is the defined as the number of its non zero coordinates. Formally:* $\|\boldsymbol{x}\| = \#\{i \text{ s.t. } x_i \neq 0\}$*. The Hamming distance between* $\boldsymbol{x}$ *and* $\boldsymbol{y}$ *is defined as* $d(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{x} - \boldsymbol{y}\|$*.*

Definition 8 (Minimum distance) *Let $\mathcal{C}$ be an $[n,k]$ linear code over $\mathbb{F}_q$. The* minimum distance *of $\mathcal{C}$ is*

$$d = \min_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{C},\boldsymbol{x}\neq\boldsymbol{y}} d(\boldsymbol{x},\boldsymbol{y}) = \min_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{C},\boldsymbol{x}\neq\boldsymbol{y}} \|\boldsymbol{x}-\boldsymbol{y}\|\,. \tag{1.7}$$

Remark 1 *Notice that by definition, the minimum distance of a linear code is exactly the minimum weight of a non-zero codeword over all possible codewords.*

When it is possible to decode, there is a unique decoding up to $\delta = \lfloor \frac{d-1}{2} \rfloor$ errors. Code parameters are denoted $[n,k,d]$.

Code-based cryptography usually suffers from huge keys. In order to keep our cryptosystem efficient, we will use the strategy of Gaborit [Gab] to decrease key sizes. This results in Quasi-Cyclic Codes, as defined below.

Definition 9 (Quasi-Cyclic Codes [MS77, Chap. 16, §7]) *View a vector $\boldsymbol{c} = (\boldsymbol{c}_0,\ldots,\boldsymbol{c}_{s-1})$ of $\mathbb{F}_2^{sn}$ as s successive blocks (n-tuples). An $[sn,k,d]$ linear code $\mathcal{C}$ is* Quasi-Cyclic (QC) *of index s if, for any $\boldsymbol{c} = (\boldsymbol{c}_0,\ldots,\boldsymbol{c}_{s-1}) \in \mathcal{C}$, the vector obtained after applying a simultaneous circular shift to every block $\boldsymbol{c}_0,\ldots,\boldsymbol{c}_{s-1}$ is also a codeword.*

More formally, by considering each block $\boldsymbol{c}_i$ as a polynomial in $\mathcal{R} = \mathbb{F}_2[X]/(X^n-1)$, the code $\mathcal{C}$ is QC of index s if for any $\boldsymbol{c} = (\boldsymbol{c}_0,\ldots,\boldsymbol{c}_{s-1}) \in \mathcal{C}$ it holds that $(X\cdot\boldsymbol{c}_0,\ldots,X\cdot\boldsymbol{c}_{s-1}) \in \mathcal{C}$.

Definition 10 (Systematic Quasi-Cyclic Codes) *A* systematic *Quasi-Cyclic $[sn,n]$ code of index s and rate $1/s$ is a quasi-cyclic code with an $(s-1)n \times sn$ parity-check matrix of the form:*

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{I}_n & 0 & \cdots & 0 & \boldsymbol{A}_0 \\ 0 & \boldsymbol{I}_n & & & \boldsymbol{A}_1 \\ & & \ddots & & \vdots \\ 0 & & \cdots & \boldsymbol{I}_n & \boldsymbol{A}_{s-2} \end{bmatrix} \tag{1.8}$$

where $\boldsymbol{A}_0,\ldots,\boldsymbol{A}_{s-2}$ are circulant $n\times n$ matrices.

Remark 2 *The definition of systematic quasi-cyclic codes of index s can of course be generalized to all rates ℓ/s, $\ell = 1\ldots s-1$, but we shall only use systematic QC-codes of rates $1/2$ and $1/3$ and wish to lighten notation with the above definition. In the sequel, referring to a systematic QC-code will imply by default that it is of rate $1/s$. Note that arbitrary QC-codes are not necessarily equivalent to a systematic QC-code.*

Definition 11 ((linear) Gilbert-Varshamov distance) *For a $[n,k]$ code over $\mathbb{F}_q$, the Gilbert-Varshamov distance is the smaller value of d which satisfies the inequality:*

$$q^{n-k} \leq \sum_{i=0}^{d} \binom{n}{i} (q-1)^i.$$

This distance corresponds to the average minimum distance of a random $[n, k]$ code. Notice that for quasi-cyclic codes, this value can be a little higher [GZ08].

Definition 12 (Words of weight w) *We define by $\mathcal{S}_w^n(\mathbb{F}_q)$ the set of all words of weight w in $\mathbb{F}_q^n$.*

1.2 Difficult problems for code-based cryptography: the Syndrome Decoding problem and its variations

The main problem considered in code-based cryptography is the Syndrome Decoding problem with different variations:

Definition 13 (Computational Syndrome Decoding (CSD) problem) *Given a random $\boldsymbol{H} \in \mathbb{F}_q^{(n-k)\times n}$, a syndrome $\mathbf{y} \in \mathbb{F}_q^{n-k}$ and an integer w.*

Question*: Is it possible to find $\boldsymbol{x} \in \mathbb{F}_q^n$ such that $\boldsymbol{H}\boldsymbol{x}^\top = \mathbf{y}$ and $\|\boldsymbol{x}\| \leq w$?*

Definition 14 (Ideal-Decision Syndrome decoding (IDSD) problem) *Given a random $\boldsymbol{H} \in \mathbb{F}_q^{(n-k)\times n}$, a syndrome $\mathbf{y} \in \mathbb{F}_q^{n-k}$ and an integer w.*

Question*: Does there exist a vector $\boldsymbol{x} \in \mathbb{F}_q^n$ such that $\boldsymbol{H}\boldsymbol{x}^\top = \mathbf{y}$ and $\|\boldsymbol{x}\| \leq w$?*

Recall that an adversary's advantage is a measure of how successfully an adversary can distinguish a cryptographic value from a idealized random value, a negligible advantage means an advantage within $O(2^{-\lambda})$ for λ the value of the security parameters in bits.

Definition 15 (Decision Syndrome Decoding problem (DSD)) *Let $\mathbf{H}$ be a random matrix $(n-k) \times n$ over $\mathbb{F}_q^n$ and let w be a positive integer.*

Question*: Given $\mathbf{x}$ of weight w, is it possible to distinguish between $\mathbf{H}\mathbf{x}^\top$ and $\mathbf{r}$ random in $\mathbb{F}_q^{n-k}$ with a non negligible advantage ?*

The Ideal-Decision Syndrome Decoding problem was proven NP-complete in [BMvT78] under the name of Coset-Weight problem, it is also called sometimes Maximum-Likelihood Decoding problem. The computational version CSD is obviously a harder problem. In practice the Ideal-Decision Syndrome Decoding problem assumes that the answer to the question can only be 'yes' or 'no', it corresponds to an ideal case, which does not fit very well the usual attacker models. In practice for security proof in cryptography, it is easier to consider a more precise notion of advantage and indistinguashability for an attacker, which motivates the introduction of the Decision Syndrome Decoding problem. The DSD problem is easier than the Ideal-DSD problem, but it was

proven in [FS96, AIK07] that over $\mathbb{F}_2$ the DSD problem was harder than the CSD problem, so that overall for random binary codes these three problems are equivalent, but in practice the DSD problem appears more naturally.

One can draw a line with the Diffie-Hellman problems, for which there is computational version CDH (Computational Diffie-Hellman) and a decisional version DDH (Decisional Diffie-Hellman), in practice proofs are often based on the DDH assumption. In the case of Diffie-Hellman problem the CDH problem is harder than the DDH problem and it is not known whether the two problems are equivalent or not.

Another related problem is the Minimum Weight problem:

Definition 16 (Minimum Weight problem) *Let $\mathcal{C}$ be a random $[n,k]$ code over $\mathbb{F}_q$, let w be a positive integer.*

Question*: Does there exist a word $\boldsymbol{x}$ in $\mathcal{C}$ of weight $\leq w$?*

This problem was proven NP-complete in [Var97].

As will be discussed in Chapter 1.4, code-based cryptography usually suffers from large public key sizes. In order to mitigate this issue, several constructions proposed to use structured code, for instance, quasi-cyclic codes. We hereafter explicitly define the syndrome decoding problem for such codes with computational and decision variations, and discuss its relative hardness.

Definition 17 (Computational s-QCSD problem) *Let $\mathcal{C}$ be a random systematic s-quasi cyclic $[sn,n]$ code over $\mathbb{F}_q$, with parity check matrix $\mathbf{H}$, and let $\boldsymbol{y}$ be a syndrome in $\mathbb{F}_q^{(s-1)n}$.*

Question: Is it possible to find $\boldsymbol{x} \in \mathbb{F}_q^{sn}$ with weight w such that $\boldsymbol{y} = \boldsymbol{H}x^\top$.

Definition 18 (Decision s-Quasi Cyclic Syndrome Decoding problem (DQCSD)) *Let $\mathbf{H}$ be a parity check matrix of a $[sn,n]$ quasi-cyclic code let w be a positive integer.*

Question*: Given $\boldsymbol{x}$ of weight w, is it possible to distinguish between $\boldsymbol{H}x^\top$ and $\mathbf{r}$ random in $\mathbb{F}_q^{n-k}$ with a non negligeable advantage ?*

There is no known reduction for the s-QCSD problem, but the problem is considered hard by the community. In the typical case for n as described in 1.4.3 there is no known attack using the quasi-cyclic structure which drastically reduces the cost of the attack for the QCSD problem. The best improvement in that case, the DOOM approach [Sen11], only permits to get a gain in $\sqrt{n}$. Concerning the reduction between the decision and the computational QCSD problems, at the difference of the previously mentionned case of random binary codes, it is not known whether the two problems are polynomially equivalent or not.

Relation between the Learning Parity with Noise (LPN) problem and the Syndrome Decoding problem

Definition 19 (Learning Parity with Noise problem (LPN)) *Fix a secret* $\mathbf{s} \in \mathbb{F}_2^n$ *and an error probability* p*. A sample* t *is defined by the binary value* $t =< r, s > +e$ *for* $\mathbf{r}$ *a random element in* $\mathbb{F}_2^n$ *and* e *an error value which is* 1 *with probability* p *and* 0 *else.*

Question*: Given N samples, is it possible to recover* s *?*

The complexity of the LPN problem depends on the number of considered samples [Lyu12]. For a fixed number of samples the LPN problem corresponds exactly to the SD problem for a $[N, n]$ binary code with an error weight depending on the probability p. To the best of our knowledge, except for the case of the Hoper-Blum (HB) authentication scheme [HB01], where the notion of unlimited samples appears naturally, for encryption schemes based on LPN, one has to fix the number of samples, the fact that the weight of the error may vary because of the probability p makes it hard to have efficient parameters, so that sometimes one considers LPN problems with fixed weight distribution. Hence, in practice efficient cryptosystems based on LPN or its variations are equivalent to cryptosystems based on the classical SD (or QCSD) problems.

1.3 Best known attacks for the Syndrome Decoding problem

Best known solvers for the syndrome decoding problem are Information Set Decoding (ISD) algorithms: a combination of linear algebra and collision search algorithms. For cryptographic applications, an adversary has to decode a noisy version $\boldsymbol{u} = \boldsymbol{c} + \boldsymbol{e}$ of a codeword $\boldsymbol{c}$, where $\boldsymbol{e}$ has (Hamming) weight w. ISD algorithms try to find an information set: a subset $\mathcal{I}$ of indexes for which $\boldsymbol{e}_i = 0$ for $i \in \mathcal{I}$, *i.e.* a error-free subset of positions. The average running time is hence a function of n, k, and w. Let $\mathcal{C}$ be a code with generator matrix $\mathbf{G}$ and let $\mathbf{H}$ be a parity check matrix associated to $\mathcal{C}$, remember that by mutiplication by a parity check matrix $\mathbf{H}$, decoding a noisy codeword $\mathbf{y} = \mathbf{mG} + \mathbf{e}$ for a small weight error $\mathbf{e}$ and a codeword $\mathbf{mG}$, is equivalent to solving $\mathbf{eH}^\top = \mathbf{yH}^\top$.

The first ISD dates back to Prange [Pra62] and is presented in Algorithm 1. The general idea consists in guessing k coordinates positions of a noisy vector and hoping that these positions do not contain errors, then if the guess is correct, one can reconstruct the word and check that the error has the small searched weight, else the attacker consider a new set of coordinates. The algorithm essentially relies on the fact that, for an invertible matrix $\boldsymbol{U} \in \mathbb{F}_2^{(n-k)\times(n-k)}$ and permutation $\boldsymbol{P} \in \mathbb{F}_2^{n\times n}$, if $\boldsymbol{H}' = \boldsymbol{UHP}$, then solving $\boldsymbol{eH}^\top = \boldsymbol{s}$ is equivalent to solving $\boldsymbol{e}'\boldsymbol{H}'^\top = \boldsymbol{s}'$, with $\boldsymbol{e}' = \boldsymbol{eP}$ and $\boldsymbol{s}' = \boldsymbol{sU}^\top$. Using Gaussian elimination, an information set can be identified in polyno-

mial time by writing the resulting parity-check matrix under systematic form. The algorithm succeeds when $\boldsymbol{sU}^\top$ has a low enough weight.

Algorithm 1: Prange-ISD

Input: $\boldsymbol{H} \in \mathbb{F}_2^{(n-k)\times n}$, $\boldsymbol{s} \in \mathbb{F}_2^{n-k}$, and target weight w
Output: $\boldsymbol{e} \in \mathbb{F}_2^n$ such that $\boldsymbol{eH}^\top = \boldsymbol{s}$ and $\|\boldsymbol{e}\| \leq w$

repeat
  Sample a uniformly random permutation $\boldsymbol{P} \in \mathbb{F}_2^{n\times n}$;
  Compute $\boldsymbol{HP}$;
  When it exists, find $\boldsymbol{U} \in \mathbb{F}_2^{(n-k)\times(n-k)}$ such that $\boldsymbol{UHP} = \left(\boldsymbol{I}_{n-k}|\tilde{\boldsymbol{H}}\right)$;
  Compute $\boldsymbol{sU}^\top$;
until $\|\boldsymbol{sU}^\top\| \leq w$;
return $\left(\boldsymbol{sU}^\top, \boldsymbol{0}\right)\boldsymbol{P}^{-1}$

The complexity of the Prange algorithm to decode an error of weight w associated to a codeword of a random $[n,k]$ code is the cost of matrix inversion times the average work factor (the inverse of the probability to find an adequate set of columns which does not intersect with the error). More precisely in the algorithm we want that the image by $\boldsymbol{P}^{-1}$ of the set of the last k positions of a vector of length n, does not contain any error position. Since $\boldsymbol{P}$ is random this probability is hence the number of possible cases for which the algorithm succeeds divided by the number of possible choices. The number of cases for which the algorithm succeeds, corresponds to the number of choices of k positions among $n-w$ positions (the total number of positions minus the w forbidden positions of the error) and the total number of choices is choosing k positions among n. The probability that the attack succeeds is therefore $\frac{\binom{n-w}{k}}{\binom{n}{k}} = \frac{\binom{n-k}{w}}{\binom{n}{w}}$ and the whole complexity of the attack is hence $O((n-k)^3\frac{\binom{n}{w}}{\binom{n-k}{w}})$. Starting from Prange [Pra62], a long line of research papers improved the complexity of these solvers [CC81, LB88, Leo88, Kro89, Ste88, CG90, vT90, Dum91, CGF91, Cha92, CC93, vT94, CC94, CC98, CS98, BLP08, BLPvT09, FS09, BLP11, MMT11, BJMM12, HS13, MO15, CS16, KT17, BM18]. As we saw with the Prange algorithm formula, the general complexity formulae depend on three parameters: n, k and the weight w. In order to compare the exponent part of the different approach one usually compares the asymptotic exponent of the attack for searching a codeword of weight the Gilbert-Varshamov bound of a $[n, n/2]$ code, which in that case is $\sim \frac{n}{9}$. For this special set of parameter, the complexity depends linearly on a unique parameter n. We give in Table 1.1 the values of the exponent depending on the attack. In the Table, the three last attacks also use Nearest Neighbor techniques together with the ISD approach, these approaches use extensive memory and are not necessarily

the most efficient for cryptographic parameters. All these attacks are probabilistic, there also exist deterministic methods but there are too expensive and in practice probabilistic methods are used. The Prange algorithm can be adapted straightforwardly to the case of finding minimum weight codewords of a code, in that case the syndrome is null, but it is possible to use the Prange algorithm by guessing an error position of the small weight vector and use the associated column of the parity-check matrix as the new syndrome.

Attacks in the case of a very small weight: in the case where w is very small compared to n and k (for instance, typically, $k = n/2$, $w = O(\sqrt{n})$), all previous attacks have the same complexity in $\sim 2^{-w \log\left(1-\frac{k}{n}\right)(1+o(1))}$ (the complexity of the original Prange algorithm) [CS16].

Quantum attacks: The best known quantum attack consists in using the Grover algorithm through the Prange algorithm and gives a complexity in $O(\sqrt{\binom{n}{w}/\binom{n-k}{w}})$ [Gro96] (see also [KT17] for small exponential improvements with other improved ISD attacks). This complexity roughly means to divide by a factor two the exponential term of the attack, and that a 256 bits classical security level (the complexity needed for an adversary to attack a problem, 2^{256} binary operations for a 256 bits security) would give a 128 quantum bits security. In contrary to number theory based problems like the factorization problem or the discrete logarithm problem, there is no known quantum attack with logarithmic gai for the Syndrome Decoding problemn, the *a priori* reason for which the community does not believe in it, is that the SD problem is NP-complete, and the general *a priori* is that this class of complexity remains hard even in front of a quantum adversary. Meanwhile the fact that there may exist a better quantum attack for the QCSD problem is still an open question.

Name	date	α
Exhaustive search		0.386
Prange [Pra62]	1962	0.1207
Stern [Ste88]	1988	0.1164
Dumer [Dum91]	1991	0.1162
May, Meurer, Thomas [MMT11]	2011	0.1114
Becker,Joux May, Meurer [BJMM12]	2012	0.1019
May, Ozerov [MO15]	2015	0.1019
Both , May [BM17]	2017	0.0953
Both, May [BM18]	2018	0.0885

Table 1.1: Evolution of the value of the coefficient α in the exponential part $2^{\alpha n}$ of the complexity of attacks for decoding a random $[n, n/2, d]$ code for $d =$ the Gilbert-Varshamov bound of the code (approx. $\frac{n}{9}$).

To gauge the gap between theory and practice, a challenge website has

KeyGen

Generate a (private) random $[n, k]$ linear code $\mathcal{C}$ and its associated generator matrix $\boldsymbol{G} \in \mathbb{F}_2^{k\times n}$ (along with an efficient decoding algorithm $\mathcal{D}$). Sample uniformly at random an invertible matrix $\boldsymbol{S} \in \mathbb{F}_2^{k\times k}$ and a permutation $\boldsymbol{P} \in \mathbb{F}_2^{n\times n}$. Return $(\mathsf{sk}, \mathsf{pk}) = \left((\boldsymbol{S}, \boldsymbol{G}, \boldsymbol{P}), \tilde{\boldsymbol{G}} = \boldsymbol{SGP}\right)$.

Encrypt

To encrypt $\boldsymbol{m} \in \mathbb{F}_2^k$, sample uniformly $\boldsymbol{e} \in \mathbb{F}_2^n$ of weight $\|\boldsymbol{e}\| = w$ and return $\boldsymbol{c} = \boldsymbol{m}\tilde{\boldsymbol{G}} + \boldsymbol{e}$.

Decrypt

To decrypt, first decode $\boldsymbol{d} = \mathcal{D}\left(\boldsymbol{cP}^{-1}\right)$, then retrieve $\boldsymbol{m}$ from by computing $\boldsymbol{dS}^{-1}$.

Why decryption works: $\mathbf{d} = (\mathbf{mS})\mathbf{G} + \mathbf{eP^{-1}}$, decoding $\mathbf{d}$ permits to recover $\mathbf{mS}$ then $\mathbf{m}$.

Figure 1.1: Generic presentation of the McEliece framework [McE78].

been recently created [ALL19]. It gathers instances of generic problems such as the SD problem or the small codeword finding problem, as well as instances more specific to NIST post-quantum cryptography standardization process.

1.4 Public-key encryption from coding theory with hidden structure

1.4.1 The McEliece and Niederreiter frameworks

The first code-based scheme was proposed by McEliece in 1978 [McE78]. This scheme is usually seen as a cryptosystem in itself, when it is more a general encryption framework, in the sense that the security reduction of the scheme depends on the family of codes considered in the instantiation. A generic description of McEliece's cryptosystem is depicted in Fig. 1.1 where sk is the **secret key** and pk the **public key**.

Parameters: the size of the public key is $(n-k)k$, the size of the ciphertext is n.

Security of the McEliece framework: The OW-CPA (One-Wayness againt Chosen Plaintext Attack) security of the scheme relies on two problems, first, the hidden code structure indistingushability from random codes which assumes that the **SGP** matrix cannot be distinguished from a random

one, then based on the previous assumption, the security is then the security of decoding random codes (hence the SD problem). From the OW-CPA security one can obtain IND-CCA2 (Indistinguishable under Chosen Plaintext Attacks) security with a small ciphertext overhead using a general conversion as for instance [HHK17].

Structural attacks the main security assumption is the indistinguishability of the public matrix from random codes. The particular type of attack called *structural attack* consists in trying to recover directly the private key $\mathbf{G}$ from the public key $\mathbf{SGP}$. There exist two type of hidden families. In a first case, the attacker knows that the public key $\mathbf{SGP}$ is obtained from a code belonging to a large family of codes but does not know the particular considered code in the large family (for instance Goppa codes). Another possibility consists in considering a unique code but which has to be resistant to recovering the primary structure from a permuted structure (for instance Reed-Muller or Reed-Solomon codes). The best generic attacks to recover a permutation from a permuted code and the code are the Leon algorithm [Leo82] (with complexity: enumerating sufficiently many small weight codewords) and the Support Splitting Algorithm [Sen00] (with complexity $\tilde{O}(2^{dim(C\cap C^{\perp})})$. In the case of specific families of codes, for instance Reed-Solomon codes with monomial permutations (a permutation where each column is moreover multiplied by a non null scalar), it is possible to break the system in polynomial time [SS92].

Niederreiter approach Niederreiter's approach which can be seen as a dual approach of McEliece, is presented in Fig. 1.2. The security is equivalent to the security of the McEliece framework, the main interest is that the ciphertext has size $(n-k)$ (the dimension of the dual) rather than n, the drawback is that the message has to be expressed in terms of small weight vector. In practice with the KEM/DEM (key encapsulation mechanism/data encapsulation mechanism) approach the latter drawback becomes less important.

Instantiations In 1978, McEliece originally proposed to instantiate this framework using binary Goppa codes $(n = 1024, k = 524, w = 50)$. The encryption and decryption procedures are fast (of complexity $O(n^2)$) compared to RSA, meanwhile the public key sizes are also large in $O(n^2)$, making it impractical compared to RSA. Other proposals were later made to either improve efficiency or reduce the key sizes [Nie86, GPT91, Sid94, JM96, BL04, Gab, BL05, BC07, BBC08, BCGO09, MB09, BLP10, MTSB13], since the framework works for any decodable code, many families of code were proposed and many were broken by structural attacks, in particular Reed-Solomon codes are an optimal family of code with a lot of structure difficult to mask. The square attack proposed in [CGG+14] is an efficient tool to distinguish codes based on codes related to Reed-Solomon codes (Reed-Solomon codes to which random columns have been added, or subcodes of Reed-Solomon codes among many possible variations) from random codes. Even if the original McEliece

cryptosystem, based on the family of Goppa codes, is still considered secure today by the communauty, the large size of public of order 260kByte (for 128 bits of security) remains a big drawback. Many variants based on alternative families of codes (Reed-Solomon codes, Reed-Muller codes or some alternant codes [MB09, BCGO09]) were broken by recovering in polynomial time the hidden structure [FOPT10]. Moreover, high rate Goppa codes have been proved not to behave like random codes [FGO+13]. The fact that the hidden code structure may be uncovered (even possibly for Goppa codes, see [COT14, FPdP14, COT17]) lies like a sword of Damocles over the system, and finding a practical alternative cryptosystem based on the difficulty of decoding unstructured or random codes has always been a major issue in code-based cryptography.

Generalized McEliece framework: the original McEliece framework described in Fig. 1.1 can be straightforwardly generalized in a scheme which can be roughly described as:

Secret Key: a decodable code $\mathcal{C}$
Public Key: a matrix $\mathbf{G}' = TrapDoor(\mathcal{C})$, for TrapDoor($\mathcal{C}$) a transformation hiding $\mathcal{C}$ in $\mathbf{G}'$.
Encryption: $\mathbf{c}=\mathbf{mG'}+\mathbf{e}$, for e a small weight error
Decryption $\mathcal{C}.Decode(TrapDoor^{-1}(c))$, for $\mathcal{C}.Decode$ a decoding algorithm of $\mathcal{C}$

This more general point of view permits not to only consider the original McEliece permutation hiding, but also more general hiding like for instance the MDPC cryptosystem or the Group-structured McEliece variations of the next sections.

This generalized framework also contains other natural variations on the scheme, for instance the Wieschbrink variation in which one adds random columns to a code to hide it [Wie06] or considering subcode of a code [BL05], or mix of the two approaches. There also exist variations [BBC+16] where the permutation is replaced by a matrix with a very small numbers of '1' in each row and column or considering monomial permutation rather than basic permutations (typically when considering non binary codes) but this type of system was also attacked [COTG15]. All these variations do not fit in the original McEliece framework, but have in common the notion of TrapDoor and hiding a decodable code in a public matrix, with a security relying on the indistinguishability of the public matrix from a random code. Most of these variations were also broken ([CLT19, LT18, BCD+16, CGG+14]).

Open questions: an interesting question would be to find a hiding TrapDoor with an proven indistinguishability, such constructions exist for lattices (see [GPV08]) but are still not known in code based cryptography. Among many broken families in the McEliece framework, it is interesting to notice that up to now there is no specific attack when considering the smaller BCH codes family rather than Goppa codes.

KeyGen

Generate a (private) random $[n, k]$ linear code $\mathcal{C}$ and its associated parity-check matrix $\boldsymbol{H}_{sec} \in \mathbb{F}_2^{(n-k)\times n}$ (along with an efficient decoding algorithm $\mathcal{D}$). Sample uniformly at random an invertible matrix $\boldsymbol{S} \in \mathbb{F}_2^{(n-k)\times(n-k)}$ and a permutation $\boldsymbol{P} \in \mathbb{F}_2^{n\times n}$. Return $(\mathsf{sk}, \mathsf{pk}) = ((\boldsymbol{S}, \boldsymbol{H}_{sec}, \boldsymbol{P}), \boldsymbol{H}_{pub} = \boldsymbol{S}\boldsymbol{H}_{sec}\boldsymbol{P})$.

Encrypt

To encrypt $\boldsymbol{m} \in \mathbb{F}_2^k$, first encode it into a vector $\tilde{\boldsymbol{m}}$ of length n and weight w, then return $\boldsymbol{c} = \boldsymbol{H}_{pub}\tilde{\boldsymbol{m}}^t$.

Decrypt

To decrypt, first decode $\boldsymbol{d} = \mathcal{D}\left(\boldsymbol{S}^{-1}\boldsymbol{c}\right)$, then retrieve $\boldsymbol{m}$ from $\tilde{\boldsymbol{m}} = \boldsymbol{d}\boldsymbol{P}^{-1}$.

Figure 1.2: Neiderreiter's cryptosystem [Nie86].

1.4.2 Group-structured McEliece framework

In order to reduce the size of the public key it is possible to consider a group-structured McEliece scheme. A quasi-cyclic McEliece framework was proposed in 2005 [Gab]. The main idea of this approach is to consider a compact representation of code through the action of a group (for instance a quasi-cyclic code) and then consider a hiding procedure compatible with the compact representation of the code, for instance considering block permutations rather than mere permutations. Quasi-cyclic codes allow to drastically reduce the size of the public key from millions of bits to a few thousand. A first approach based on quasi-cyclic BCH codes was broken in [OTD10], an improved approach based on alternant codes [BCGO09] had their proposed parameters broken [FOPT10] eventually this approach led to the BigQuake submission (based on quasi-cyclic Goppa codes) [BBB+17b] to the NIST competition, whose parameters have never been attacked and permit to gain a factor roughly 10 on the classical McEliece Goppa instantiation. Another approach based on similar idea but with the action of another group (the dyadic group) [MB09] had some of their parameters broken in [FOPT10] and led to the NIST submission [BBB+17a] whose parameters were broken in [BC18]. Overall this approach should lead to a decrease of the size of the public key by some factor, now the underlying quasi-cyclic structure may be seen as a weakness.

Security and parameters: everything is similar to the McEliece framework except that the size of the public is smaller by a factor roughly 10 and more structure is added on the hidden code,the reduction is done with the QCSD problem and not with the SD problem.

1.4.3 Moderate-Density Parity-Check codes

At last the MDPC cryptosystem introduced in 2013 in [MTSB13] uses a hidden code approach as for the McEliece framework, but in that case no hiding permutation is used, the masking comes from the knowledge of small weight codewords of the code which permit to decode the MDPC code.

Since LDPC are a very efficient class of decodable codes it is very natural to consider them within the McEliece framework. However the very reason why these codes are efficient is the fact that the dual matrix is built from very small weight vectors and this is the precisely why they cannot be used as such in the McEliece framework, since recovering very small words in the dual is easy.

A first reasonable approach to use LDPC codes for cryptography was done in [BBC08] through a masking of the LDPC structure, this type of approach will eventually lead to the LEDA submission to the NIST standarization process [BBC+17], eventually attacked in [DA20].

The main idea from [MTSB13] is to consider a generalization of LDPC codes: LDPC codes have very small weight vectors of weight $O(1)$ in their dual but decode in $O(n)$, Moderate Parity Check codes will be generated from higher weight codewords in $O(\sqrt{n})$ but will also be less efficient and decode errors of weight $O(\sqrt{n})$. Now, when LDPC codes have been studied for a long time, MDPC codes still raise questions regarding the analysis of their decoding rate, especially in a cryptographic context for which a Decryption Failure Rate (DFR) in 2^{-128} is expected.

In order to get smaller size of key, it is also possible to consider quasi-cyclic version of MDPC codes (QC-MDPC codes). The main advantage of QC-MDPC codes is that the knowledge of one small weight vector is enough to decode them. In practice the QC-MDPC cryptosystem (usually simply denoted by MDPC although the codes are quasi-cyclic) can be seen as being close to the NTRU (N^{th} degree Truncated polynomial Ring Units) cryptosystem but in Hamming weight version. The basic decoding algorithm presented in Algo. 2, based on the classical LDPC decoder, is very simple to describe (the value of the threshold T plays a capitol role), but usually optimized variations are considered [SV19, DGK20b]. The original scheme was described in a McEliece like version, we describe in Fig. 1.3, a polynomial key exchange version similar to what is presented in [AAB+17a] (see also Section 1.4.3).

Notice that turning the key exchange protocol into an encryption scheme is straightforward.

Algorithm 2: BitFlipping$(\boldsymbol{h}_0, \boldsymbol{h}_1, \boldsymbol{s}, T, t)$

Input: $\boldsymbol{h}_0, \boldsymbol{h}_1$, and $\boldsymbol{s} = \boldsymbol{h}_1\boldsymbol{e}_1 + \boldsymbol{h}_0\boldsymbol{e}_0$, threshold value T required to flip a bit, weight t of $\boldsymbol{e}_0$ and $\boldsymbol{e}_1$.

Output: $(\boldsymbol{e}_0, \boldsymbol{e}_1)$ if the algorithm succeeds, $\perp$ otherwise.

$(\boldsymbol{u}, \boldsymbol{v}) \leftarrow (\boldsymbol{0}, \boldsymbol{0}) \in (\mathbb{F}_2^n)^2$, $\boldsymbol{H} \leftarrow \left(\mathbf{rot}(-\boldsymbol{h}_0)^\top, \mathbf{rot}(\boldsymbol{h}_1)^\top\right) \in \mathbb{F}_2^{n\times 2n}$, syndrome $\leftarrow \boldsymbol{s}$;

while $[\|\boldsymbol{u}\| \neq t$ **or** $\|\boldsymbol{v}\| \neq t]$ **and** $\|$syndrome$\| \neq 0$ **do**

  sum $\leftarrow$ syndrome$\times\boldsymbol{H}$; `/* No modular reduction, values in` $\mathbb{Z}$ `*/`

  flipped_positions $\leftarrow \boldsymbol{0} \in \mathbb{F}_2^{2n}$;

  for $i \in [\![0, 2n-1]\!]$ **do**

    if sum$[i] \geq T$ **then**

      flipped_positions$[i]$ = flipped_positions$[i] \oplus 1$;

  $(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{u}, \boldsymbol{v}) \oplus$ flipped_positions;

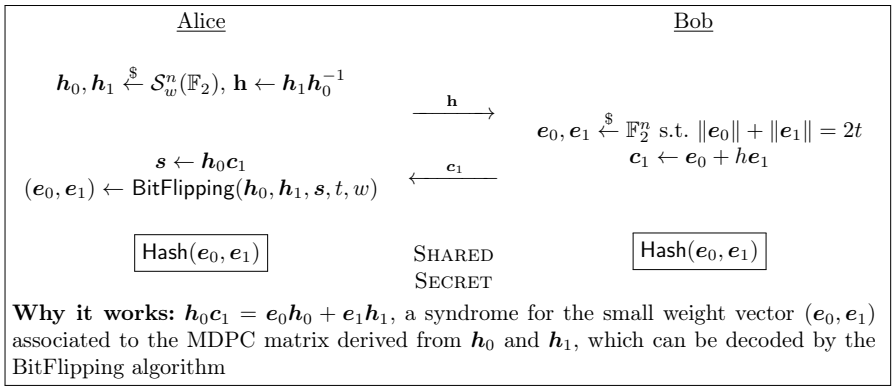
  syndrome = syndrome $-\boldsymbol{H} \times$ flipped_positions$^\top$;

if $\boldsymbol{s} - \boldsymbol{H} \times (\boldsymbol{u}, \boldsymbol{v})^\top \neq 0$ **then**

  return $\perp$;

else

  return $(\boldsymbol{u}, \boldsymbol{v})$;



Why it works: $\boldsymbol{h}_0\boldsymbol{c}_1 = \boldsymbol{e}_0\boldsymbol{h}_0 + \boldsymbol{e}_1\boldsymbol{h}_1$, a syndrome for the small weight vector $(\boldsymbol{e}_0, \boldsymbol{e}_1)$ associated to the MDPC matrix derived from $\boldsymbol{h}_0$ and $\boldsymbol{h}_1$, which can be decoded by the BitFlipping algorithm

Figure 1.3: Description of QC-MDPC Key Exchange protocol in a Niederreiter form, n usually follows the typical case described in 1.4.3, w is odd, h_0 and h_1 are small weight vectors considered as elements of $\mathcal{R}$, h_0 is taken invertible in $\mathcal{R}$, the products follows the definition of eq. 1.3, with simplified notation xy rather than $x \cdot y$. An encryption version can be obtained by considering h as a public key and adding a second part in Bob's response, $\boldsymbol{c}_0 = \mathbf{m}$ XOR $Hash(\boldsymbol{e}_0, \boldsymbol{e}_1)$, w and t are of weight $O(\sqrt{n})$).

Parameters: for security level λ, the length of the code is in $O(\lambda^2)$ and the weight of generating vector is in $O(\lambda)$.

Security: the original presentation of the MDPC scheme had an OW-CPA security based on the QCSD problem, but also on the indistinguishabilty between the public matrix of a QC-MDPC (generated by a small weight vector) and a random quasi-cyclic code. The Niederreiter version presented in Fig. 1.3 follows [AAB+17a] and can be proven IND-CCA2 (under the two previous hypothesis) with a generic transformation of type [HHK17] at the condition that the DFR is proven sufficiently low in $2^{-\lambda}$, for λ the security parameter in bits.

Attacks and DFR: the main attacks are related to the decryption failure rate [GJS16],the MDPC scheme can be used for key exchange in a one-time process with a DFR sufficiently small as 2^{-30}, using it for encryption necessitates a stronger analysis of the decoder to reach a DFR of at least 2^{-128} [SV19]. A strong drawback of the cryptosystem for one-time key exchange was the cost of inverting a polynomial, which is drastically improved in [DGK20a].

Variations on the scheme the original MDPC scheme [MTSB13] is in a McEliece like form, it is possible to consider Niederreiter like form like for the BIKE-2 version of the NIST standardization process, and also a version wich does not necessitate a costly inversion [BGG+17] (the BIKE-1 version of the NIST standardization process), see the Bike submission [AAB+17a] for details.

1.5 PKE schemes with reduction to decoding random codes without hidden structure

1.5.1 Alekhnovich's approach

In 2003, Alekhnovich proposed an innovative approach based on the difficulty of decoding purely random codes [Ale03]. In this system the trapdoor (or secret key) is a random error vector that has been added to a random codeword of a random code. Recovering the secret key is therefore equivalent to solving the problem of decoding a random code – with no hidden structure. Alekhnovich also proved that breaking the system in any way, not necessarily by recovering the secret key, involves decoding a random linear code. The single bit encryption scheme is presented in Fig. 1.4. The scheme can be turned into a multiple bits encryption scheme and necessitates the use of error-correcting codes to handle the decryption failure rate induced by the system. Typically one encrypts a codeword (seen as a sequence of single bits) obtained as an encoding of the plaintext by a given decodable code, the decryption results in a noisy codeword which can be decoded to recover the codeword and then from it, the plaintext.

KeyGen

Let $\boldsymbol{A} \xleftarrow{\$} \mathbb{F}_2^{k\times n}$, $\boldsymbol{x} \xleftarrow{\$} \mathbb{F}_2^k$, $w = O(\sqrt{n})$, $\boldsymbol{e} \xleftarrow{\$} \mathcal{S}_w^n(\mathbb{F}_2)$, $\boldsymbol{y} \leftarrow \boldsymbol{x}\boldsymbol{A} + \boldsymbol{e}$ and $\boldsymbol{H} \leftarrow \left(\boldsymbol{A}^\top \mid \boldsymbol{y}^\top\right)^\top$. Let $\mathcal{C}$ be the code with parity-check matrix $\boldsymbol{H}$, and $\boldsymbol{G}$ be a generator matrix for $\mathcal{C}$. Return $(\mathsf{sk}, \mathsf{pk}) = (\boldsymbol{e}, \boldsymbol{G})$.

Encrypt

To encrypt $m \in \{0, 1\}$, if $m = 0$, $\boldsymbol{c} \leftarrow \boldsymbol{c}' + \boldsymbol{e}'$, else $\boldsymbol{c} \leftarrow \boldsymbol{u}$, for $\boldsymbol{c}'$ a random codeword of $\mathcal{C}$, $\boldsymbol{e}' \xleftarrow{\$} \mathcal{S}_w^n$, and $\boldsymbol{u} \xleftarrow{\$} \mathbb{F}_2^n$. Return $\boldsymbol{c}$.

Decrypt

To decrypt, return $b \leftarrow \langle \boldsymbol{c}, \boldsymbol{e} \rangle$.

Why decryption works: if $m = 0$, $\langle \boldsymbol{c}, \boldsymbol{e} \rangle = \langle \boldsymbol{c}' + \boldsymbol{e}', \boldsymbol{e} \rangle = \langle \boldsymbol{e}', \boldsymbol{e} \rangle = 0$ with overwhelming probability (since $w = O(\sqrt{n})$). If $m = 1$, decryption succeeds with probability $1/2$. The scheme is probabilistic.

Figure 1.4: Alekhnovich encryption scheme for a single bit [Ale03].

Even if the system was not totally practical, the approach in itself was a breakthrough for code-based cryptography. Its inspiration was provided in part by the Ajtai-Dwork cryptosystem [AD97] which is based on solving hard lattice problems. The Ajtai-Dwork cryptosystem also inspired the Learning With Errors (LWE) lattice-based cryptosystem by Regev [Reg03] which generated a huge amount of work in lattice-based cryptography.

1.5.2 HQC: Efficient encryption from random quasi-cyclic Codes

The previous Alekhnovich cryptosystem, although not quite efficient was a first step towards efficient cryptosystems based on random code. In 2018, Aguilar *et al.* proposed an efficient cryptosystem based on the difficulty of decoding random quasi-cyclic codes and on the notion of noisy Diffie-Hellman (see also [GC10] for the first version of the scheme in 2010). The main novelty of the system is that at the difference of the McEliece framework, there is not only one masked code which does both encryption AND decryption (through decoding), but two codes: a first random double circulant code which is used for encryption and a second code which is used for decryption. The novelty in the scheme is that this second code is public with no masking, it just needs to have an efficient decoding algorithm. In particular considering other decoding codes does not change the security of the scheme. The scheme is very simple and expressed in polynomial form (see Section 1.4.3 for the relation between polynomial representation and matrices) and described in Fig. 1.5. The Hamming Quasi-Cyclic (HQC) scheme was submitted to the NIST standardization process [AAB+17c].

Parameters: similarly to MDPC codes, for λ the security level, the length of the code is in $O(\lambda^2)$ and the weight of the small weight generating vectors is in $O(\lambda)$. In practice the parameters are slighlty larger than for MDPC and two blocks are needed for the ciphertext.

Security: the IND-CPA security of HQC relies on the QCSD problem and the decision DQCSD problem, no indistinguashability hypothesis for a hidden code is needed. The IND-CCA2 security can be reached through a generic and efficient transformation such as [HHK17] depending on the DFR of the system.

Instantiations: in the original HQC submission, the use of a tensor code obtained from BCH codes and a repetition code was proposed for decoding. Recently a more efficient concatenated code based on Reed-Muller and Reed-Solomon codes was proposed in [AGZ20]. Typically one needs to decode codes with a very low rate of order 1% and an error rate of order 30%.

Decryption Failure Rate analysis: the main point to have in mind for this system is that since the decoding is probabilistic, one needs to have a precise DFR analysis for the security proof which requires a DFR in $2^{-\lambda}$ for λ the security level, which is the case with proposed decoding code. One could consider more efficient iterative decoding algorithms which exist in the litterature, but in that case having a precise DFR analysis which is able to go as a low as 2^{-128} is trickier to obtain.

A relative weakness of the system is its relatively low encryption rate, but this is not a major issue for classical applications of public-key encryption schemes such as authentication or key exchange, in the NIST standardization process, it was possible to consider a message size of only 256 bits.

- $\mathsf{KeyGen}(\mathsf{param})$: samples $\boldsymbol{h} \xleftarrow{\$} \mathcal{R}$, a generator matrix $\boldsymbol{G} \in \mathbb{F}_2^{k\times n}$ of $\mathcal{C}$ which decodes $O(n)$ errors, $\mathsf{sk} = (\boldsymbol{x}, \boldsymbol{y}) \xleftarrow{\$} \mathcal{R}^2$ such that $\|\boldsymbol{x}\| = \|\boldsymbol{y}\| = O(\sqrt{n})$, sets $\mathsf{pk} = (\boldsymbol{h}, \boldsymbol{s} = \boldsymbol{x} + \boldsymbol{hy})$, and returns $(\mathsf{pk}, \mathsf{sk})$.
- $\mathsf{Encrypt}(\mathsf{pk}, \boldsymbol{m})$: generates $\boldsymbol{e} \xleftarrow{\$} \mathcal{R}$, $\boldsymbol{r} = (\boldsymbol{r}_1, \boldsymbol{r}_2) \xleftarrow{\$} \mathcal{R}^2$ such that $\|\boldsymbol{e}\| = w_{\boldsymbol{e}}$, and $\|\boldsymbol{r}_1\| = \|\boldsymbol{r}_2\| = w_{\boldsymbol{r}}$, for $w_{\boldsymbol{r}} = w_{\boldsymbol{e}} = O(\sqrt{n})$. Sets $\boldsymbol{u} = \boldsymbol{r}_1 + \boldsymbol{hr}_2$ and $\boldsymbol{v} = \boldsymbol{mG} + \boldsymbol{sr}_2 + \boldsymbol{e}$, returns $\boldsymbol{c} = (\boldsymbol{u}, \boldsymbol{v})$.
- $\mathsf{Decrypt}(\mathsf{sk}, \boldsymbol{c})$: returns $\mathcal{C}.\mathsf{Decode}(\boldsymbol{v} - \boldsymbol{uy})$.

Why decryption works: $\boldsymbol{v} - \boldsymbol{uy} = \mathbf{mG} + \mathbf{yr_1} + \mathbf{xr_2} + \mathbf{e}$, a word decodable by $\mathcal{C}$, with a decodable error of weight $O(n)$ for chosen parameters.

Figure 1.5: Description of the HQC cryptosystem. The multiplication $\mathbf{xy}$ of two elementis $\mathbf{x}$ and $\mathbf{y}$ of $\mathcal{R}$ is a simplified notation of $\mathbf{x} \cdot \mathbf{y}$ of eq. 1.3, the choice of n follows the typical case of 1.4.3.

1.5.3 Ouroboros key exchange protocol

While HQC has the advantage of relying only on the hardness of the (decisional) syndrome decoding problem for random quasi-cyclic codes, it still features big keys for low encryption rates. For most concrete applications such as key exchange over internet through TLS for instance, having a small plaintext space, big enough to encrypt a session key is sufficient.

Based upon this observation, lighter parameters (and some other tricks) were proposed for a key-exchange protocol, resulting in the Ouroboros key exchange protocol [DGZ17] and presented in Fig. 1.6. The protocol works similarly as HQC, except that no message needs to be encrypted (hence there is no public code $\mathcal{C}$), and the decoding algorithm, which is very close to the Bit Flipping algorithm [Gal63] used for LDPC and MDPC codes, is tweaked to handle noisy syndromes (see Algorithm 3). The Ouroboros protocol was submitted to the NIST standardization process as BIKE-3.

Security and parameters: the Ouroboros protocol can be seen as intermediate to HQC and MDPC schemes: it benefits from the security reduction of HQC with reduction to attacking random quasi-cyclic instances on one side (the QCSD and DQCSD problems), and one the other side it uses an extended bitflip-like decoding algorithm (xBitFlipping). As for the HQC protocol, the Ouroboros protocol necessitates to send two blocks for the encryption. Similarly to MDPC and HQC, for λ the security level, the length of the code is in $O(\lambda^2)$ and the weight of the small weight generating vector is in $O(\lambda)$. In practice the parameters lie between those of HQC and MDPC. Notice that since the value f_1 is random, a seed to generate it is sufficient. As for MDPC, the Ouroboros key exchange protocol can be turned into an encryption scheme which can be proven IND-CCA2 through a generic transformation like [HHK17] depending on a low DFR, but without additional public matrix indistinguishability hypothesis.

Decryption Failure Rate analysis: the decoding of Ouroboros is very close to the decoding of MDPC codes, the properties of the decoder are very close in practice. Basically for a small DFR as 2^{-30}, like MDPC, the Ouroboros protocol can be used as a KEM for key exchange, it can be use for encryption but with a lower DFR.

Key exchange and encryption similarly to MDPC the protocol is pre-

sented as a key exchange protocol but it can be easily turned into an encryption scheme for a message $\mathbf{m}$, by adding a ciphertext $c = \mathbf{m}$ XOR $Hash(e_0, e_1)$.

Algorithm 3: xBitFlipping$(\boldsymbol{h}_0, \boldsymbol{h}_1, \boldsymbol{s}, T, t)$

Input: $\boldsymbol{h}_0, \boldsymbol{h}_1$, and $\boldsymbol{s} = \boldsymbol{h}_1\boldsymbol{e}_1 - \boldsymbol{h}_0\boldsymbol{e}_0 + \boldsymbol{e}$, a threshold value T required to flip a bit, weight t of $\boldsymbol{e}_0$, $\boldsymbol{e}_1$ and $\boldsymbol{e}$.

Output: $(\boldsymbol{e}_0, \boldsymbol{e}_1)$ if the algorithm succeeds, $\perp$ otherwise.

$(\boldsymbol{u}, \boldsymbol{v}) \leftarrow (\mathbf{0}, \mathbf{0}) \in (\mathbb{F}_2^n)^2$, $\boldsymbol{H} \leftarrow \left(\mathbf{rot}(-\boldsymbol{h}_0)^\top, \mathbf{rot}(\boldsymbol{h}_1)^\top\right) \in \mathbb{F}_2^{n \times 2n}$, syndrome $\leftarrow \boldsymbol{s}$;

while $[\|\boldsymbol{u}\| \neq t$ **or** $\|\boldsymbol{v}\| \neq t]$ **and** $\|$syndrome$\| > t$ **do**

  sum $\leftarrow$ syndrome$\times\boldsymbol{H}$; `/* No modular reduction, values in` $\mathbb{Z}$ `*/`

  flipped_positions $\leftarrow \mathbf{0} \in \mathbb{F}_2^{2n}$;

  for $i \in [\![0, 2n-1]\!]$ **do**

    if sum$[i] \geq T$ **then**

      flipped_positions$[i]$ = flipped_positions$[i] \oplus 1$;

  $(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{u}, \boldsymbol{v}) \oplus$ flipped_positions;

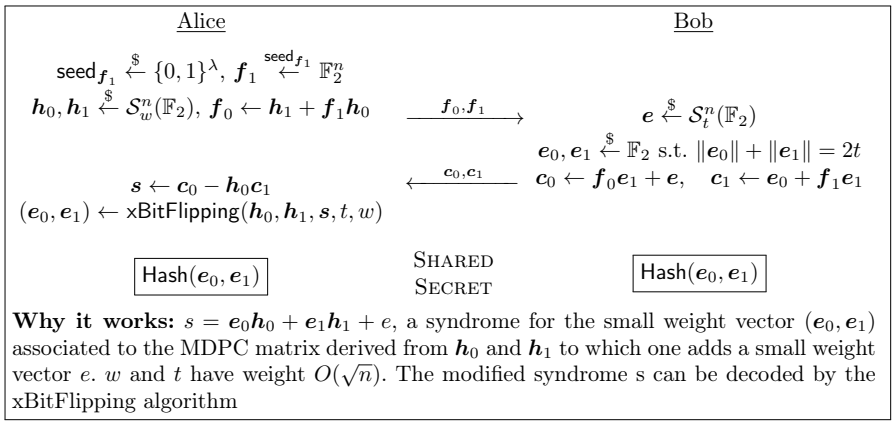
  syndrome = syndrome $-\boldsymbol{H} \times$ flipped_positions$^\top$;

if $\left\|\boldsymbol{s} - \boldsymbol{H} \times (\boldsymbol{u}, \boldsymbol{v})^\top\right\| > t$ **then**

  return $\perp$;

else

  return $(\boldsymbol{u}, \boldsymbol{v})$;



Why it works: $s = \boldsymbol{e}_0\boldsymbol{h}_0 + \boldsymbol{e}_1\boldsymbol{h}_1 + e$, a syndrome for the small weight vector $(\boldsymbol{e}_0, \boldsymbol{e}_1)$ associated to the MDPC matrix derived from $\boldsymbol{h}_0$ and $\boldsymbol{h}_1$ to which one adds a small weight vector e. w and t have weight $O(\sqrt{n})$. The modified syndrome s can be decoded by the xBitFlipping algorithm

Figure 1.6: Description of the Ouroboros Key Exchange protocol. $\boldsymbol{f}_0$ and $\boldsymbol{f}_1$ constitute the public key. Alternatively $\boldsymbol{f}_1$ can be recovered by sending/publishing only the λ bits of the seed $seed_{f_1}$ (instead of the n coordinates of $\boldsymbol{f}_1$). The choice of n follows the typical case of 1.4.3.

1.6 Examples of parameters for code-based encryption and key exchange

In this section we provide a table with practical examples for 128 bits security.

Cryptosystem name	pk size	ct size	DFR
Classic McEliece	261120	128	0
Big Quake	25482	201	0
HQC-original	3024	6017	2^{-128}
HQC-RMRS	2607	5191	2^{-128}
BIKE-II (MDPC)	1473	1505	2^{-128}
BIKE-III (Ouroboros)	1566	3100	2^{-128}
BIKE-I (Cake)	2945	2945	2^{-128}

Table 1.2: Comparison between several 1^{st} and 2^{nd} round code-based submissions to NIST Post-Quantum Cryptography (PQC) standardization process. All sizes are expressed in bytes. The targeted security level is NIST Level-1, approximately equivalent to 128 bits of classical security. Notation *ct* stands for ciphertext.

1.7 Authentication: the Stern zero-knowledge protocol

Authentication is an important cryptographic primitive. When there is an obvious generic transformation to turn a public key encryption schemes or a signature scheme into a public key authentication scheme, there also exist specific authentication schemes and in particular zero-knowledge authentication schemes. After the seminal work of Fiat and Shamir there were a series of papers related to code-based authentication schemes which culminated in efficiency with the Stern authentication algorithm.

We now recall Stern's zero-knowledge[2] authentication protocol [Ste93]. Given a random public parity-check matrix $\boldsymbol{H} \in \mathbb{F}_2^{(n-k)\times n}$, the secret key is a small weight vector $\boldsymbol{x}$ of length n and of weight w. The public key is constructed as $\boldsymbol{s} = \boldsymbol{H}\boldsymbol{x}^\top$. In Stern's identification protocol, a prover P wants

[2]Often due to the original Stern's protocol description, the protocol is presented without the random seeds r_1, r_2 and r_3, in that case the protocol is not Zero-Knowledge but only Testable Weak Zero knowledge [ABCG17], random seeds $\boldsymbol{r}_i$ need to be added like in Fig 1.7.

to convince a verifier V that he is indeed the person corresponding to the public identifier $\boldsymbol{s}$.

Secret key: $\boldsymbol{x} \in \mathbb{F}_2^n$ of weight w
Public key: a random $(n-k) \times n$ binary matrix $\mathbf{H}, \mathbf{s} = \mathbf{H}\mathbf{x}^\top$

1. [Commitment Step] P samples uniformly at random a commitment $\boldsymbol{y} \in \mathbb{F}_2^n$, a permutation σ of $\{1, 2, \ldots, n\}$ and three random seeds $\boldsymbol{r}_i$ for $i \in \{1, 2, 3\}$ of $\mathbb{F}_2^\lambda$, for λ the security level in bits. Then P sends to V the commitments $\boldsymbol{c}_1$, $\boldsymbol{c}_2$ and $\boldsymbol{c}_3$ such that :

$$\boldsymbol{c}_1 = h(\boldsymbol{r}_1|\sigma|\boldsymbol{H}\boldsymbol{y}^\top);\ \boldsymbol{c}_2 = h(\boldsymbol{r}_2|\sigma(\boldsymbol{y}));\ \boldsymbol{c}_3 = h(\boldsymbol{r}_3|\sigma(\boldsymbol{y}+\boldsymbol{x})),$$

   where $a|b$ denotes the concatenation of a and b.

2. [Challenge Step] V sends $b \in \{0, 1, 2\}$ to P.

3. [Answer Step] Three possibilities :

   - if $b = 0$: P reveals $\boldsymbol{r}_1$, $\boldsymbol{r}_2$, $\boldsymbol{y}$ and σ.
   - if $b = 1$: P reveals $\boldsymbol{r}_1$, $\boldsymbol{r}_3$, $(\boldsymbol{y} + \boldsymbol{x})$ and σ.
   - if $b = 2$: P reveals $\boldsymbol{r}_2$, $\boldsymbol{r}_3$, $\sigma(\boldsymbol{y})$ and $\sigma(\boldsymbol{x})$.

4. [Verification Step] Three possibilities :

   - if $b = 0$: V verifies that $\boldsymbol{c}_1, \boldsymbol{c}_2$ have been honestly calculated.
   - if $b = 1$: V verifies that $\boldsymbol{c}_1, \boldsymbol{c}_3$ have been honestly calculated.
   - if $b = 2$: V verifies that $\boldsymbol{c}_2, \boldsymbol{c}_3$ have been honestly calculated, and that the weight of $\sigma(\boldsymbol{x})$ is w.

5. Iterate the steps 1,2,3,4 until the expected security level is reached.

Why it works: for b=1, $\mathbf{H}\mathbf{y}^\mathbf{t}$ can be obtained as $\mathbf{H}\mathbf{y}^\mathbf{t} = \mathbf{H}(\mathbf{x}+\mathbf{y})^\mathbf{t} + \mathbf{s}$

Figure 1.7: Stern's protocol

Stern's protocol has a cheating probability of 2/3 for each round. This implies that the protocol has to be repeated $\lceil -\lambda / \log_2(2/3) \rceil$ times to achieve a negligible (in the security parameter λ) cheating probability. For instance, for $\lambda = 128$ bits of security, this results in 219 rounds.

Security and parameters: the security of the protocol is the generic SD problem, there are two main drawbacks to the scheme: the public key, a random matrix, is very large and the cheating probability is rather high, which implies a large number of round to get small overall cheating probabilities. In particular the signature is very large and is on the order of the square of the security level, indeed increasing the security level increases both the

length of the considered code and the number of necessary rounds. Typically for parameters one considers a secret word of weight just below the Gilbert-Varshamov bound for a $1/2$ rate code. It leads to lengths of order 1000 for the code for 128 bits security.

Variations and improvements: there are two main improvements on the scheme, first the introduction of quasi-cyclicity to decrease the size of the public matrix in [GG07] and second the cheating probability which can be decreased to $\frac{1}{2}$ when considering quasi-cyclic codes [AGS11]. There exists a dual version of the protocol slightly more efficient in [Vér95]. It is also possible to obtain a $\frac{1}{2}$ cheating probability by considering q-ary codes [CVA10]. An interesting question would be to try to decrease the cheating probablity for one round, below the $\frac{1}{2}$ best known cheating probability.

1.8 Digital signatures from coding theory

Designing a secure and efficient signature scheme based on coding theory is a long-standing open problem. A first approach consists in designing a zero-knowledge identification scheme and then turn it into a digital signature scheme by applying the Fiat-Shamir transform to identification transcript, using a collision-resistant hash function h (not necessarily based on coding assumptions). This kind of approach usually yields large signatures that are unpractical for real-life use.

The other approach is known as the hash-and-sign paradigm and will be described later in this section. While this latter approach can yield much shorter and efficient signature schemes, it is rather hard to obtain securely.

1.8.1 Signature from zero-knowledge authentication scheme with the Fiat-Shamir Heuristic

The Fiat-Shamir heuristic permits to turn a zero-knowledge protocol into a signature scheme, the main idea is that the new protocol is no longer interactive anymore and the prover proves itself through the ise of a hash function. We recall the Fiat-Shamir heuristic in Fig. 1.8.

Security: the Fiat-Shamir heuristic is proven secure in the random oracle model ([PS96, ADV+12]), the security is based on the SD problem for the original Stern algorithm and under QCSD for its quasi cyclic improvement. The question of the extension of the proof in a quantum oracle is actively considered by researchers.

- KeyGen(param): Keygen parameters of the zero-knowledge scheme, returns (pk, sk).
- Signature of a message M: Fix a security a level and a number N of associated rounds. Generate the N comitments C_i at once, and compute the general comitment C as a concatenation of all the C_i. Compute a sequence of N challenges r_i from a hash of the concatenation of C and the message M. Compute the N answers A_i associated to the r_i and concatenate the A_i as A. The signature is the two supersets (C,A).
- Verification: Compute the sequence of challenges r_i from the received C and M. Check that the answers A_i corresponds to the challenges r_i and comitments C_i. Accept the signature if all the answers fit else reject.

Figure 1.8: Signature through the Fiat-Shamir heuristic.

Parameters : the main advantage of such signatures is that the public key is rather small (in the quasi-cyclic version), a few hundred bits, and the security is really strong. Now the signature length in itself is rather large (a few hundred thousand bits).

Variations and applications : because of the simplicity of the scheme and the really strong security of the scheme, Fiat-Shamir heuristic based signature have been used to introduce many of the classical signatures used in classical number theory based cryptography: blind signatures [OB17], group signatures [ELL+15], undeniable signatures [ABGS13], ring signatures [ZLC, ACGL11], identity-based signatures [CGG07], etc....

1.8.2 the CFS signature scheme

A classical way to build a hash-and-sign signature, like the RSA signature for instance, is to consider a full domain hash in which it is possible to associate to any hash (an element of a domain) a pre-image through a particular hard to invert function. For coding theory this approach is basically somehow difficult, indeed if one considers the whole space $\mathbb{F}_2^n$ and a given code $\mathcal{C}$ the set of decodable codewords is exactly the whole space $\mathbb{F}_2^n$ if and only if the code is perfect. There are not so many such codes and typically their structure is hard to hide. The idea of the CFS scheme proposed by Courtois Finiasz and Sendrier in [CFS01], is to consider a classical McEliece scheme built with Goppa codes, but then to consider very dense Goppa codes in order to obtain a high density of decodable words. The algorithm consists then in considering hash values obtained from the message to sign together with a counter, and

check whether the hash value considered as a syndrome can be decoded, and repeat it until one obtains a decodable hash value.

For a t-correcting binary Goppa code $\mathcal{C}$ of length $n = 2^m$ over $\mathbb{F}_2$, only $\binom{n}{t}$ among 2^{mt} syndromes are decodable, yielding an asymptotic density of $\frac{\binom{n}{t}}{2^{mt}} \simeq \frac{1}{t!}$.

The scheme is described in Fig. 1.9, all details can be found in the original paper [CFS01].

Security: The security proof of the scheme relies on the indistinguishability of a dense permuted Goppa code. It was proven in [FGO+13] that this assumption did not hold and that it was possible to distinguish between dense Goppa codes and random codes, this property holds for very dense Goppa codes but not for Goppa codes used in the classical encryption scheme for which the rate of the code is often of order $1/2$. It is interesting to notice that even of it is possible to distinguish between dense Goppa codes and random codes, in practice there is no known attack on the scheme from this distinguisher.

Parameters: the density in $\frac{1}{t!}$ obliges to consider on the average $t!$ trials, hence t cannot be too large, moreover decoding t errors for a random code has to be difficult. Overall these two constraints lead to considering very long Goppa codes so that the size of the public key is super-polynomial relatively to the security level. In practice the scheme can be used for not too high level of security like 80 bits of security (in that case $n = 2^{16}, n-k = 144$ and $t = 9$), higher security levels like 128 bits seem out of reach in practice. Meanwhile since for a long time the scheme was the only existing hash-and-sign signature, it was used in many code-based constructions.

KeyGen

Use Niederreiter KeyGen algorithm (see Fig. 1.2) to generate a public parity check matrix $\boldsymbol{H}_{pub}$ and a secret parity check matrick $\boldsymbol{H}_{sec}$.

Sign

Compute $\boldsymbol{s}_i \leftarrow h\left(h\left(\boldsymbol{m}\right)|i\right)$ until $\boldsymbol{s}_{i_0}$ can be decoded in $\boldsymbol{z}$ using $\boldsymbol{H}_{sec}$. The signature is $(\boldsymbol{z}, i_0)$.

Verify

Accept the signature if $\boldsymbol{H}_{pub}\boldsymbol{z}^\top = h\left(h\left(\boldsymbol{m}\right)|i_0\right)$. Reject otherwise.

Figure 1.9: CFS signature scheme

1.8.3 the WAVE signature

At last recently a new hash-and-sign signature was proposed in [DST19], this scheme considers a ternary generalized $(\mathbf{U}, \mathbf{U}+\mathbf{V})$-code construction an

extension of the $(\mathbf{u}, \mathbf{u}+\mathbf{v})$ construction of Definition **??**. This construction is in the spirit of the lattice construction [GPV08] in the sense that a proof is given that the signature is indistinguishable from a random distribution by using rejection sampling. Moreover in contrary to the CFS scheme, for which in the range of parameters considered, there is a unique pre-image for considered syndromes, in the case of WAVE, the trapdoor is used to construct a pre-image of a syndrome but in a range where the number of preimages is not unique. At last the paper introduces a very interesting new notion of decoding for high weights. The security of the scheme relies both on the hardness of generic decoding for high weights and the indistinguishability of generalized (U,U+V)-codes. Even if the parameters are rather large, they do not increase as dramatically as the CFS signature scheme and hence the WAVE signature scheme can be considered as the first hash-and-sign code-based signature scheme. For 128 bits of classical security, signature sizes are in the order of 15 thousand bits, the public key size in the order of 4 megabytes, and the rejection rate is limited to one rejection every 10 to 12 signatures.

1.8.4 Few-times signatures schemes and variations

One of the main problem to design a signature scheme is the fact that there has to be no leak of information when a signature is given. Some schemes exist which permit to obtain only a few signatures. The main idea of these schemes is to give as public key, a set of syndromes associated to small weight secret vectors. The signature is then a preimage (build with the secret vectors) of a linear combination of the public syndromes. The first proposed scheme was the Kabatianskii–Krouk–Smeets (KKS) scheme wrongly proposed at first as being a general signature scheme (see [KKS97] and attacks [OT11]), also more recently proposed full signature schemes adaptating ideas from lattices were also broken or too much inefficient [LKLN17, DG19] eventually becoming few-times signatures schemes. A few-times signature scheme based on the action of automorphism was also presented in [GS12].

1.9 Other primitives

There exist many other cryptographic primitives, not all of them possess code-based equivalent, but some do.

Code-based Pseudo-Random Number Generators: it is possible to construct pseudo-random generators based on the SD problem, the first scheme was proposed in [FS96], a more efficient variation based on quasi-cyclic codes was proposed in [GLS07].

Code-based hash functions: hash function are a very important tool in cryptography, following what was done for lattices, a first code-based hash was proposed in [AFS05], it was followed by a quasi-cyclic optimized versions [AFG+08, FGS07], submitted to the SHA3 (Secure Hash Algorithm 3) competition in 2008, and other variations [BLPS11].

Open question on primitives: among primitives which are still not known and of real interest one can cite Homomorphic Encryption, it is trivially possible to add $O(n)$ ciphertexts by a code-based cryptosystem and still be able to decrypt the sum of them, but it is not known if it is possible with $O(n^2)$ ciphertext or even more. Also it is not known how to obtain Fully Homomorphic Encryption. At last Identity-based Encryption is also an open question.

1.10 Rank-based cryptography

Rank-based cryptography is based on codes in the rank metric rather than Hamming metric. The rank distance consists in considering codes over an extension F_{q^m}, a codeword v over $F_{q^m}^n$ can be unfolded in a $\mathbb{F}_q$ basis of F_{q^m} as a $m \times n$ matrix V, the rank weight of v is then the rank of the matrix V. Alternatively the rank of a word v, is the dimension of the $\mathbb{F}_q$ space generated by its coordinates. There is a natural dictionnary of many notion in Hamming distance when they are considered with rank metric. For instance the notion of permutation is turned into a notion of invertible matrix on the base field, and the notion of support in Hamming is turned into the space generated by coordinates in rank metric. Also the SD problem has an analog in rank metric: the Rank Syndrome Decoding problem (RSD) where the Hamming distance is replaced by the rank metric. The fact, changing the metric implies several changes on geometrical properties of the codes. Overall some cryptographic schemes can be easiliy adapted, for instance the GPT cryptosystem [GPT91] is an adaptation of the McEliece cryptosystem with Gabidulin codes replacing Goppa codes. Gabidulin codes have though a very strong structure hard to mask which lead to many attack on the scheme or its variations [Ove05]. There also exist equivalent systems to MDPC, the LRPC (Low Rank Parity Check) cryptosystem [GMRZ13], to HQC: the RQC (Rank Quasi Cyclic) cryptosystem [ABD+18] or to Ouroboros scheme, the Ouroboros-R scheme [AAB+17b], these schemes were presented in the ROLLO 2nd round submission of the NIST standardization process [ABD+19].

After a first attempt of signature, the RankSign signature [GRSZ14] which was attacked in [DT18], an efficient signature Durandal was proposed recently in [ABG+19].

It is possible to adapt the Stern authentication scheme [GSZ11], other

primitives as [GHT16] are also known, but some other like hash functions seem to resist because of properties of the metric.

More generally rank metric leads in general to cryptosystems with smaller size of keys. The general difficulty of the problem is not yet completely fixed even if recent years have permitted to develop a better knowledge of the security of the general Rank Syndrome Decoding problem ([GRS16, BBC+20] to which the Syndrome Decoding (SD) problem can be probabilistically reduced to ([GZ16]).

Acknowledgement

The first author thanks Olivier Blazy, Alain Couvreur, Thomas Debris-Alazard and Gilles Zémor for helpful comments and discussions.

Bibliography

[AAB+17a] Carlos Aguilar Melchor, Nicolas Aragon, Paulo Barreto, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Shay Gueron, Tim Güneysu, Rafael Misoczki, Edoardo Persichetti, Nicolas Sendrier, Jean-Pierre Tillich, and Gilles Zémor. BIKE. First round submission to the NIST post-quantum cryptography call, November 2017. *14, 16*

[AAB+17b] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loic Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, and Gilles Zémor. Ouroboros-R. First round submission to the NIST post-quantum cryptography call, November 2017. *27*

[AAB+17c] Carlos Aguilar Melchor, Nicolas Aragon, Slim Bettaieb, Loïc Bidoux, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Edoardo Persichetti, and Gilles Zémor. HQC, November 2017. NIST Round 1 submission for Post-Quantum Cryptography. *17*

[ABCG17] Quentin Alamélou, Olivier Blazy, Stéphane Cauchie, and Philippe Gaborit. A code-based group signature scheme. *Des. Codes Cryptogr*, 82(1-2):469–493, 2017. *21*

[ABD+18] Carlos Aguilar Melchor, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, and Gilles Zémor. Efficient encryption from random quasi-cyclic codes. *IEEE Trans. Inform. Theory*, 64(5):3927–3943, 2018. *27*

[ABD+19] Nicolas Aragon, Olivier Blazy, Jean-Christophe Deneuville, Philippe Gaborit, Adrien Hauteville, Olivier Ruatta, Jean-Pierre Tillich, Gilles Zémor, Carlos Aguilar Melchor, Slim Bettaieb, Loïc Bidoux, Bardet Magali, and Ayoub Otmani. ROLLO (merger of Rank-Ouroboros, LAKE and LOCKER). Second round submission to the NIST post-quantum cryptography call, March 2019. *27*

[ABG+19] Nicolas Aragon, Olivier Blazy, Philippe Gaborit, Adrien Hauteville, and Gilles Zémor. Durandal: a rank metric based

signature scheme. In *Advances in Cryptology - EUROCRYPT 2019 - 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19-23, 2019, Proceedings, Part III*, volume 11478 of *LNCS*, pages 728–758. Springer, 2019. *27*

[ABGS13] Carlos Aguilar Melchor, Slim Bettaieb, Philippe Gaborit, and Julien Schrek. A code-based undeniable signature scheme. In Martijn Stam, editor, *14th IMA International Conference on Cryptography and Coding*, volume 8308 of *LNCS*, pages 99–119. Springer, Heidelberg, December 2013. *24*

[ACGL11] Carlos Aguilar Melchor, Pierre-Louis Cayrel, Philippe Gaborit, and Fabien Laguillaumie. A new efficient threshold ring signature scheme based on coding theory. *IEEE Trans. Inform. Theory*, 57(7):4833–4842, 2011. *24*

[AD97] Miklós Ajtai and Cynthia Dwork. A public-key cryptosystem with worst-case/average-case equivalence. In *Proceedings of the Twenty-Ninth Annual ACM Symposium on the Theory of Computing, El Paso, Texas, USA, May 4-6, 1997*, pages 284–293, 1997. *17*

[ADV+12] Sidi Mohamed El Yousfi Alaoui, Özgür Dagdelen, Pascal Véron, David Galindo, and Pierre-Louis Cayrel. Extended security arguments for signature schemes. In Aikaterini Mitrokotsa and Serge Vaudenay, editors, *AFRICACRYPT 12*, volume 7374 of *LNCS*, pages 19–34. Springer, Heidelberg, July 2012. *23*

[AFG+08] Daniel Augot, Matthieu Finiasz, Philippe Gaborit, Stéphane Manuel, and Nicolas Sendrier. SHA-3 proposal : FSB. In *Submission to the SHA3 NIST competition,*, 2008. *27*

[AFS05] Daniel Augot, Matthieu Finiasz, and Nicolas Sendrier. A family of fast syndrome based cryptographic hash functions. In *Ed Dawson, Serge Vaudenay (editors). Progress cryptology-Mycrypt First international conference on cryptology Malaysia, ISBN 978-3-540-28938-8*, volume 3715 of *LNCS*, pages 64–83, Kuala Lumpur, Malaysia, September 2005. Springer. *27*

[AGS11] Carlos Aguilar, Philippe Gaborit, and Julien Schrek. A new zero-knowledge code based identification scheme with reduced communication. In *Proc. IEEE Inf. Theory Workshop- ITW 2011*, pages 648–652. IEEE, October 2011. *23*

[AGZ20] Nicolas Aragon, Philippe Gaborit, and Gilles Zémor. Hqc-rmrs, an instantiation of the hqc encryption framework with a more efficient auxiliary error-correcting code. https://arxiv.org/abs/2005.10741, 2020. *18*

[AIK07] Benny Applebaum, Yuval Ishai, and Eyal Kushilevitz. Cryptography with constant input locality. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 92–110. Springer, Heidelberg, August 2007. *6*

[Ale03] Michael Alekhnovich. More on average case vs approximation complexity. In *44th FOCS*, pages 298–307. IEEE Computer Society Press, October 2003. *16*, *17*

[ALL19] Nicolas Aragon, Julien Lavauzelle, and Matthieu Lequesne. decodingchallenge.org, 2019. http://decodingchallenge.org. *10*

[BBB+17a] Gustavo Banegas, Paulo S.L.M Barreto, Brice Odilon Boidje, Pierre-Louis Cayrel, Gilbert Ndollane Dione, Kris Gaj, Cheikh Thiécoumba Gueye, Richard Haeussler, Jean Belo Klamti, Ousmane N'diaye, Duc Tri Nguyen, Edoardo Persichetti, and Jefferson E. Ricardini. DAGS : Key encapsulation for dyadic GS codes. https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/DAGS.zip, November 2017. First round submission to the NIST post-quantum cryptography call. *13*

[BBB+17b] Magali Bardet, Élise Barelli, Olivier Blazy, Rodolfo Canto Torres, Alain Couvreur, Phillipe Gaborit, Ayoub Otmani, Nicolas Sendrier, and Jean-Pierre Tillich. BIG QUAKE. https://bigquake.inria.fr, November 2017. NIST Round 1 submission for Post-Quantum Cryptography. *13*

[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. Springer-Verlag, 2008. *11*, *14*

[BBC+16] Marco Baldi, Marco Bianchi, Franco Chiaraluce, Joachim Rosenthal, and Davide Schipani. Enhanced public key security for the McEliece cryptosystem. *J. Cryptology*, 29(1):1–27, 2016. *12*

[BBC+17] Marco Baldi, Alessandro Barenghi, Franco Chiaraluce, Gerardo Pelosi, and Paolo Santini. LEDAkem. First round submission to the NIST post-quantum cryptography call, November 2017. *14*

[BBC+20] Magali Bardet, Maxime Bros, Daniel Cabarcas, Philippe Gaborit, Ray Perlner, Daniel Smith-Tone, Jean-Pierre Tillich, and Javier Verbel. Algebraic attacks for solving the Rank Decoding and MinRank problems without Gröbner basis. *arXiv e-prints*, page arXiv:2002.08322, Feb 2020. *28*

[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. *11*

[BC18] Élise Barelli and Alain Couvreur. An efficient structural attack on NIST submission DAGS. In Thomas Peyrin and Steven Galbraith, editors, *Advances in Cryptology - ASIACRYPT'18*, volume 11272 of *LNCS*, pages 93–118. Springer, December 2018. *13*

[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 2016*, LNCS, pages 118–143, Fukuoka, Japan, February 2016. *12*

[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 *LNCS*, pages 77–97, Gammarth, Tunisia, June 21-25 2009. *11*, *12*, *13*

[BGG+17] Paulo S. L. M. Barreto, Shay Gueron, Tim Güneysu, Rafael Misoczki, Edoardo Persichetti, Nicolas Sendrier, and Jean-Pierre Tillich. CAKE: code-based algorithm for key encapsulation. In *Cryptography and Coding - 16th IMA International Conference, IMACC 2017, Oxford, UK, December 12-14, 2017, Proceedings*, volume 10655 of *LNCS*, pages 207–226. Springer, 2017. *16*

[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*, LNCS. Springer, 2012. *8*, *9*

[BL04] Thierry P. Berger and Pierre Loidreau. Designing an efficient and secure public-key cryptosystem based on reducible rank codes. In *Progress in Cryptology - INDOCRYPT 2004*, volume 3348 of *LNCS*, pages 218–229, 2004. *11*

[BL05] Thierry P. Berger and Pierre Loidreau. How to mask the structure of codes for a cryptographic use. *Des. Codes Cryptogr.*, 35(1):63–79, 2005. *11*, *12*

[BLP08] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Attacking and defending the McEliece cryptosystem. In *Post-Quantum Cryptography 2008*, volume 5299 of *LNCS*, pages 31–46, 2008. *8*

[BLP10] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Wild McEliece. In Alex Biryukov, Guang Gong, and Douglas R. Stinson, editors, *Selected Areas in Cryptography*, volume 6544 of *LNCS*, pages 143–158, 2010. *11*

[BLP11] Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Smaller decoding exponents: ball-collision decoding. In *Advances in Cryptology - CRYPTO 2011*, volume 6841 of *LNCS*, pages 743–760, 2011. *8*

[BLPS11] Daniel J. Bernstein, Tanja Lange, Christiane Peters, and Peter Schwabe. Really fast syndrome-based hashing. In *Progress in Cryptology - AFRICACRYPT 2011*, volume 6737 of *LNCS*, pages 134–152. Springer, 2011. *27*

[BLPvT09] D. J. Bernstein, T. Lange, C. Peters, and H. van Tilborg. Explicit bounds for generic decoding algorithms for code-based cryptography. In *Pre-proceedings of WCC 2009*, pages 168–180, 2009. *8*

[BM17] Leif Both and Alexander May. Optimizing BJMM with Nearest Neighbors: Full Decoding in $2^{2/21n}$ and McEliece Security. In *WCC Workshop on Coding and Cryptography*, September 2017. available at http://cits.rub.de/imperia/md/content/may/paper/bjmm+.pdf. *9*

[BM18] Leif Both and Alexander May. Decoding linear codes with high error rate and its impact for LPN security. In Tanja Lange and Rainer Steinwandt, editors, *Post-Quantum Cryptography 2018*, volume 10786 of *LNCS*, pages 25–46, Fort Lauderdale, FL, USA, April 2018. Springer. *8, 9*

[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. *5*

[CC81] GC Jr Clark and JB Cain. Error-correction coding for digital communications. *New York, Plenum Press, 1981. 434 p.*, 1981. *8*

[CC93] Hervé Chabanne and Bernard Courteau. Application de la méthode de décodage itérative d'Omura a la cryptanalyse du système de McEliece. Technical Report 122, University of Sherbrooke, 1993. *8*

[CC94] Anne Canteaut and Hervé Chabanne. A further improvement of the work factor in an attempt at breaking McEliece's cryptosystem. In *EUROCODE 94*, pages 169–173. INRIA, 1994. *8*

[CC98] Anne Canteaut and Florent Chabaud. A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511. *IEEE Trans. Inform. Theory*, 44(1):367–378, 1998. *8*

[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 *LNCS*, pages 157–174, Gold Coast, Australia, 2001. Springer. *24*, *25*

[CG90] John T Coffey and Rodney M Goodman. The complexity of information set decoding. *IEEE Transactions on Information Theory*, 36(5):1031–1037, 1990. *8*

[CGF91] John T Coffey, Rodney M Goodman, and Patrick G Farrell. New approaches to reduced-complexity decoding. *Discrete Applied Mathematics*, 33(1-3):43–60, 1991. *8*

[CGG07] Pierre-Louis Cayrel, Philippe Gaborit, and Marc Girault. Identity-based identification and signature schemes using correcting codes. In Daniel Augot, Nicolas Sendrier, and Jean-Pierre Tillich, editors, *WCC 2007*, pages 69–78. INRIA, 2007. *24*

[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. *Des. Codes Cryptogr.*, 73(2):641–666, 2014. *11*, *12*

[Cha92] Florent Chabaud. Asymptotic analysis of probabilistic algorithms for finding short codewords. In Sami Harari Paul Camion, Pascale Charpin, editor, *Eurocode '92. Proceedings of the International Symposium on Coding Theory and Applications*, pages 175–183, Udine, Italy, October 1992. Springer. *8*

[CLT19] Alain Couvreur, Matthieu Lequesne, and Jean-Pierre Tillich. Recovering short secret keys of RLCE in polynomial time. In Jintai Ding and Rainer Steinwandt, editors, *Post-Quantum Cryptography 2019*, volume 11505 of *LNCS*, pages 133–152, Chongquing, China, May 2019. Springer. *12*

[COT14] 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 *LNCS*, pages 17–39. Springer Berlin Heidelberg, 2014. *12*

[COT17] Alain Couvreur, Ayoub Otmani, and Jean-Pierre Tillich. Polynomial time attack on wild McEliece over quadratic extensions. *IEEE Trans. Inform. Theory*, 63(1):404–427, Jan 2017. *12*

[COTG15] Alain Couvreur, Ayoub Otmani, Jean-Pierre Tillich, and Valérie Gauthier-Umaña. A polynomial-time attack on the BBCRS scheme. In J. Katz, editor, *Public-Key Cryptography - PKC 2015*, volume 9020 of *LNCS*, pages 175–193. Springer, 2015. *12*

[CS98] Anne Canteaut and Nicolas Sendrier. Cryptanalysis of the original McEliece cryptosystem. In *Advances in Cryptology - ASIACRYPT 1998*, volume 1514 of *LNCS*, pages 187–199. Springer, 1998. *8*

[CS16] Rodolfo Canto-Torres and Nicolas Sendrier. Analysis of information set decoding for a sub-linear error weight. In *Post-Quantum Cryptography 2016*, LNCS, pages 144–161, Fukuoka, Japan, February 2016. *8, 9*

[CVA10] Pierre-Louis Cayrel, Pascal Véron, and Sidi Mohamed El Yousfi Alaoui. A zero-knowledge identification scheme based on the q-ary syndrome decoding problem. In *Selected Areas in Cryptography*, pages 171–186, 2010. *23*

[DA20] Angela Robinson Paolo Santini Daniel Apon, Ray A. Perlner. Cryptanalysis of ledacrypt. IACR Cryptol. ePrint Arch. 2020: 455, 2020. *14*

[DG19] Jean-Christophe Deneuville and Philippe Gaborit. Cryptanalysis of a code-based one-time signature. In *to appear in Desi. Codes and Crypt.*, 2019. *26*

[DGK20a] Nir Drucker, Shay Gueron, and Dusan Kostic. Fast polynomial inversion for post quantum qc-mdpc cryptography. In *CSCML 2020*, pages 110–127, 2020. *16*

[DGK20b] Nir Drucker, Shay Gueron, and Dusan Kostic. QC-MDPC decoders with several shades of gray. In Jintai Ding and Jean-Pierre Tillich, editors, *PQCrypto 2020*, volume 12100 of *LNCS*, pages 35–50. Springer, 2020. *14*

[DGZ17] Jean-Christophe Deneuville, Philippe Gaborit, and Gilles Zémor. Ouroboros: A simple, secure and efficient key exchange protocol based on coding theory. In *Post-Quantum Cryptography - 8th International Workshop, PQCrypto 2017, Utrecht, The Netherlands, June 26-28, 2017, Proceedings*, volume 10346 of *LNCS*, pages 18–34. Springer, 2017. *19*

[DST19] Thomas Debris-Alazard, Nicolas Sendrier, and Jean-Pierre Tillich. Wave: A new family of trapdoor one-way preimage sampleable functions based on codes. In *Advances in Cryptology - ASIACRYPT 2019*, LNCS, Kobe, Japan, December 2019. Springer. *25*

[DT18] Thomas Debris-Alazard and Jean-Pierre Tillich. Two attacks on rank metric code-based schemes: Ranksign and an identity-based-encryption scheme. In *Advances in Cryptology - ASIACRYPT 2018*, volume 11272 of *LNCS*, pages 62–92, Brisbane, Australia, December 2018. Springer. *27*

[Dum91] Ilya Dumer. On minimum distance decoding of linear codes. In *Proc. 5th Joint Soviet-Swedish Int. Workshop Inform. Theory*, pages 50–52, Moscow, 1991. *8, 9*

[ELL+15] Martianus Frederic Ezerman, Hyung Tae Lee, San Ling, Khoa Nguyen, and Huaxiong Wang. A provably secure group signature scheme from code-based assumptions. In Tetsu Iwata and Jung Hee Cheon, editors, *ASIACRYPT 2015, Part I*, volume 9452 of *LNCS*, pages 260–285. Springer, Heidelberg, November / December 2015. *24*

[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. *12, 25*

[FGS07] Matthieu Finiasz, Philippe Gaborit, and Nicolas Sendrier. Improved Fast Syndrome Based Cryptographic Hash Functions. In V. Rijmen, editor, *ECRYPT Hash Workshop 2007*, 2007. *27*

[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 *LNCS*, pages 279–298, 2010. *12, 13*

[FPdP14] Jean-Charles Faugère, Ludovic Perret, and Frédéric de Portzamparc. Algebraic attack against variants of McEliece with Goppa polynomial of a special form. In *Advances in Cryptology - ASIACRYPT 2014*, volume 8873 of *LNCS*, pages 21–41, Kaoshiung, Taiwan, R.O.C., December 2014. Springer. *12*

[FS96] Jean-Bernard Fischer and Jacques Stern. An efficient pseudo-random generator provably as secure as syndrome decoding. In Ueli Maurer, editor, *Advances in Cryptology - EUROCRYPT'96*, volume 1070 of *LNCS*, pages 245–255. Springer, 1996. *6, 26*

[FS09] Matthieu Finiasz and Nicolas Sendrier. Security bounds for the design of code-based cryptosystems. In M. Matsui, editor, *Advances in Cryptology - ASIACRYPT 2009*, volume 5912 of *LNCS*, pages 88–105. Springer, 2009. *8*

[Gab] Philippe Gaborit. In *Proceedings of the 2005 International Workshop on Coding and Cryptography (WCC 2005), Book of extended abstracts*, pages 81–91, Bergen, Norway, March. https://www.unilim.fr/pages_perso/philippe.gaborit/shortIC.ps. *4*, *11*, *13*

[Gal63] Robert G. Gallager. *Low Density Parity Check Codes*. M.I.T. Press, Cambridge, Massachusetts, 1963. *19*

[GC10] Philippe Gaborit and Aguilar Carlos Melchor. Cryptographic method for communicating confidential information. https://patents.google.com/patent/EP2537284B1/en, 2010. *17*

[GG07] Philippe Gaborit and Marc Girault. Lightweight code-based authentication and signature. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 191–195, Nice, France, June 2007. *23*

[GHT16] Philippe Gaborit, Adrien Hauteville, and Jean-Pierre Tillich. Ranksynd a PRNG based on rank metric. In *Post-Quantum Cryptography 2016*, pages 18–28, Fukuoka, Japan, February 2016. *28*

[GJS16] Qian Guo, Thomas Johansson, and Paul Stankovski. A key recovery attack on MDPC with CCA security using decoding errors. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *Advances in Cryptology - ASIACRYPT 2016*, volume 10031 of *LNCS*, pages 789–815, 2016. *16*

[GLS07] Philippe Gaborit, Cédric Lauradoux, and Nicolas Sendrier. SYND: a fast code-based stream cipher with a security reduction. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, pages 186–190, Nice, France, June 2007. *26*

[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. *27*

[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 LNCS, pages 482–489, Brighton, April 1991. *11*, *27*

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 197–206. ACM, 2008. *12*, *26*

[Gro96] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proc. 28th Annual ACM Symposium on the Theory of Computation*, pages 212–219, New York, NY, 1996. ACM Press, New York. 1, 9

[GRS12] Venkatesan Guruswami, Atri Rudra, and Madhu Sudan. Essential coding theory. *Draft available at https://cse.buffalo.edu/faculty/atri/courses/coding-theory/book/web-coding-book.pdf*, 2012. 3

[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. 28

[GRSZ14] Philippe Gaborit, Olivier Ruatta, Julien Schrek, and Gilles Zémor. Ranksign: An efficient signature algorithm based on the rank metric (extended version on arxiv). In *Post-Quantum Cryptography 2014*, volume 8772 of *LNCS*, pages 88–107. Springer, 2014. 27

[GS12] Philippe Gaborit and Julien Schrek. Efficient code-based one-time signature from automorphism groups with syndrome compatibility. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT 2012*, pages 1982–1986, Cambridge, MA, USA, July 2012. 26

[GSZ11] Philippe Gaborit, Julien Schrek, and Gilles Zémor. Full cryptanalysis of the chen identification protocol. In *Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011, Taipei, Taiwan, November 29 - December 2, 2011. Proceedings*, pages 35–50, 2011. 27

[GZ08] Philippe Gaborit and Gilles Zémor. Asymptotic improvement of the Gilbert-Varshamov bound for linear codes. *IEEE Trans. Inform. Theory*, 54(9):3865—3872, 2008. 2, 5

[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. 28

[HB01] Nicholas J. Hopper and Manuel Blum. Secure human identification protocols. In Colin Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 52–66. Springer, Heidelberg, December 2001. 7

[HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In *Theory of Cryptography Conference*, pages 341–371. Springer, 2017. 11, 16, 18, 19

[HP10] W Cary Huffman and Vera Pless. *Fundamentals of error-correcting codes*. Cambridge university press, 2010. *3*

[HS13] Yann Hamdaoui and Nicolas Sendrier. A non asymptotic analysis of information set decoding. IACR Cryptology ePrint Archive, Report2013/162, 2013. http://eprint.iacr.org/2013/162. *8*

[JM96] Heeralal Janwa and Oscar Moreno. McEliece public key cryptosystems using algebraic-geometric codes. *Des. Codes Cryptogr.*, 8(3):293–307, 1996. *11*

[KKS97] Gregory Kabatianskii, Evgenii Krouk, and Ben. J. M. Smeets. A digital signature scheme based on random error-correcting codes. In *IMA Int. Conf.*, volume 1355 of *LNCS*, pages 161–167. Springer, 1997. *26*

[Kro89] Evgenii Avramovich Krouk. Decoding complexity bound for linear block codes. *Problemy Peredachi Informatsii*, 25(3):103–107, 1989. *8*

[KT17] Ghazal Kachigar and Jean-Pierre Tillich. Quantum information set decoding algorithms. In *Post-Quantum Cryptography 2017*, volume 10346 of *LNCS*, Utrecht, The Netherlands, June 2017. Springer. *8*, *9*

[LB88] Pil J. Lee and Ernest F. Brickell. An observation on the security of McEliece's public-key cryptosystem. In *Advances in Cryptology - EUROCRYPT'88*, volume 330 of *LNCS*, pages 275–280. Springer, 1988. *8*

[Leo82] Jeffrey Leon. Computing automorphism groups of error-correcting codes. *IEEE Trans. Inform. Theory*, 28(3):496–511, 1982. *11*

[Leo88] Jeffrey Leon. A probabilistic algorithm for computing minimum weights of large error-correcting codes. *IEEE Trans. Inform. Theory*, 34(5):1354–1359, 1988. *8*

[LKLN17] Wijik Lee, Young-Sik Kim, Yong-Woo Lee, and Jong-Seon No. Post quantum signature scheme based on modified Reed-Muller code pqsigRM. First round submission to the NIST post-quantum cryptography call, November 2017. *26*

[LT18] Matthieu Lequesne and Jean-Pierre Tillich. Attack on the edon-kkey encapsulation mechanism. In *2018 IEEE International Symposium on Information Theory, ISIT 2018, Vail, CO, USA, June 17-22, 2018*, pages 981–985, 2018. *12*

[Lyu12] Vadim Lyubashevsky. Lattice signatures without trapdoors. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 738–755. Springer, Heidelberg, April 2012. *7*

[MB09] Rafael Misoczki and Paulo Barreto. Compact McEliece keys from Goppa codes. In *Selected Areas in Cryptography*, Calgary, Canada, August 13-14 2009. *11, 12, 13*

[McE78] Robert J. McEliece. *A Public-Key System Based on Algebraic Coding Theory*, pages 114–116. Jet Propulsion Lab, 1978. DSN Progress Report 44. *1, 10*

[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 *LNCS*, pages 107–124. Springer, 2011. *8, 9*

[MO15] Alexander May and Ilya Ozerov. On computing nearest neighbors with applications to decoding of binary linear codes. In E. Oswald and M. Fischlin, editors, *Advances in Cryptology - EUROCRYPT 2015*, volume 9056 of *LNCS*, pages 203–228. Springer, 2015. *8, 9*

[MS77] Florence Jessie MacWilliams and Neil James Alexander Sloane. *The theory of error-correcting codes*, volume 16. Elsevier, 1977. *3, 4*

[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. *11, 14, 16*

[Nie86] Harald Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problems of Control and Information Theory*, 15(2):159–166, 1986. *1, 11, 13*

[NIS17] NIST. Post-quantum cryptography standardization. https://csrc.nist.gov/Projects/post-quantum-cryptography/Post-Quantum-Cryptography-Standardization, 2017. *1*

[OB17] Julien Schrek Nicolas Sendrier Olivier Blazy, Philippe Gaborit. A code-based blind signature. In *ISIT 2017*, pages 2718–2722, 2017. *24*

[OT11] Ayoub Otmani and Jean-Pierre Tillich. An efficient attack on all concrete KKS proposals. In *Post-Quantum Cryptography 2011*, volume 7071 of *LNCS*, pages 98–116, 2011. *26*

[OTD10] Ayoub Otmani, Jean-Pierre Tillich, and Léonard Dallot. Cryptanalysis of two McEliece cryptosystems based on quasi-cyclic codes. *Special Issues of Mathematics in Computer Science*, 3(2):129–140, January 2010. *13*

[Ove05] Raphael Overbeck. A new structural attack for GPT and variants. In *Mycrypt*, volume 3715 of *LNCS*, pages 50–63, 2005. *27*

[Pra62] Eugene Prange. The use of information sets in decoding cyclic codes. *IRE Transactions on Information Theory*, 8(5):5–9, 1962. *7, 8, 9*

[PS96] David Pointcheval and Jacques Stern. Security proofs for signature schemes. In Ueli M. Maurer, editor, *EUROCRYPT'96*, volume 1070 of *LNCS*, pages 387–398. Springer, Heidelberg, May 1996. *23*

[Reg03] Oded Regev. New lattice based cryptographic constructions. In *35th ACM STOC*, pages 407–416. ACM Press, June 2003. *17*

[Sen00] Nicolas Sendrier. Finding the permutation between equivalent linear codes: The support splitting algorithm. *IEEE Trans. Inform. Theory*, 46(4):1193–1203, 2000. *11*

[Sen11] Nicolas Sendrier. Decoding one out of many. In *Post-Quantum Cryptography 2011*, volume 7071 of *LNCS*, pages 51–67, 2011. *6*

[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. *1*

[Sid94] Vladimir Michilovich Sidelnikov. A public-key cryptosytem based on Reed-Muller codes. *Discrete Math. Appl.*, 4(3):191–207, 1994. *11*

[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. *11*

[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 *LNCS*, pages 106–113. Springer, 1988. *8, 9*

[Ste93] Jacques Stern. A new identification scheme based on syndrome decoding. In D.R. Stinson, editor, *Advances in Cryptology - CRYPTO'93*, volume 773 of *LNCS*, pages 13–21. Springer, 1993. *21*

[SV19] Nicolas Sendrier and Valentin Vasseur. On the decoding failure rate of QC-MDPC bit-flipping decoders. In Jintai Ding and Rainer Steinwandt, editors, *Post-Quantum Cryptography 2019*, volume 11505 of *LNCS*, pages 404–416, Chongquing, China, May 2019. Springer. *14, 16*

[Var97] Alexander Vardy. The intractability of computing the minimum distance of a code. *IEEE Trans. Inform. Theory*, 43(6):1757–1766, November 1997. *6*

[Vér95] Pascal Véron. A fast identification scheme. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT*, page 359, September 1995. *23*

[vT90] Johan van Tilburg. On the McEliece public-key cryptosystem. In *Advances in Cryptology - CRYPTO'88*, volume 403 of *LNCS*, pages 119–131, London, UK, 1990. Springer. *8*

[vT94] Johan van Tilburg. *Security-analysis of a class of cryptosystems based on linear error-correcting codes.* PhD thesis, Technische Universiteit Eindhoven, 1994. *8*

[Wie06] 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. *12*

[ZLC] Dong Zheng, Xiangxue Li, and Kefei Chen. *International Journal of Network Security.* *24*