RankSign : an efficient signature algorithm based on the rank metric

Philippe Gaborit[1], Olivier Ruatta[1],Julien Schrek[1] and Gilles Zémor[2]

[1] Université de Limoges, XLIM-DMI,
123, Av. Albert Thomas
87060 Limoges Cedex, France.
gaborit,schrek,ruatta@unilim.fr
[2] Université de Bordeaux,
Institut de Mathématiques, UMR 5251,
zemor@math.u-bordeaux1.fr

Abstract. In this paper we propose a new approach to code-based signatures that makes use in particular of rank metric codes. When the classical approach consists in finding the unique preimage of a syndrome through a decoding algorithm, we propose to introduce the notion of mixed decoding of erasures and errors for building signature schemes. In that case the difficult problem becomes, as is the case in lattice-based cryptography, finding a preimage of weight above the Gilbert-Varshamov bound (case where many solutions occur) rather than finding a unique preimage of weight below the Gilbert-Varshamov bound. The paper describes RankSign: a new signature algorithm for the rank metric based on a new mixed algorithm for decoding erasures and errors for the recently introduced Low Rank Parity Check (LRPC) codes. We explain how it is possible (depending on choices of parameters) to obtain a full decoding algorithm which is able to find a preimage of reasonable rank weight for any random syndrome with a very strong probability. We study the semantic security of our signature algorithm and show how it is possible to reduce the unforgeability to direct attacks on the public matrix, so that no information leaks through signatures. Finally, we give several examples of parameters for our scheme, some of which with public key of size 5760 bits and signature of size 1728 bits. Moreover the scheme can be very fast for small base fields.

Keys words: Signature algorithm, rank metric

1 Introduction

The problem of finding an efficient signature algorithm has been a major challenge for code-based cryptography since its introduction in 1978 by McEliece. Signing with error-correcting codes can be achieved in different ways: the CFS algorithm [7] considers extreme parameters of Goppa codes to obtain a class of codes in which a non-negligeable part of random syndromes are invertible. This scheme has a very small signature size, however it is rather slow and the public key is very large. Another possibility is to use the Fiat-Shamir heuristic to turn a zero-knowledge authentication scheme (like the Stern authentication scheme [28]) into a signature scheme. This approach leads to very small public keys of a few hundred bits and is rather fast, but the signature size in itself is large (about 100,000b), so that overall no wholly satisfying scheme is known.

Besides classical code-based cryptography that relies on the Hamming distance, it is also possible to use another metric: the rank metric. This metric introduced in 1985 by Gabidulin [11]

is very different from the Hamming distance. The rank metric has received in recent years very strong attention from the coding community for their application to network coding. Moreover, this metric can also be used for cryptography. Indeed it is possible to construct rank-analogues of Reed-Solomon codes: the Gabidulin codes. Gabidulin codes inspired early cryptosystems, like the GPT cryposystem ([12]), but they turned out to be inherently vulnerable because of the very strong structure of the underlying codes. More recently, by considering an approach similar to NTRU [18](and also MDPC codes [24]) constructing a very efficient cryptosystem based on weakly structured rank codes was shown to be possible [13]. However, in terms of signatures based on the rank metric, only systems that use Fiat-Shamir are presently known [14].

An interesting point in code-based cryptography is that in general the security of the protocols relies on finding small weight vectors *below* the Gilbert-Varshamov bound (the typical minimum weight of a random code). This is noticeably different from lattice based cryptography for which it is very common for the security of a signature algorithm [17, 23] to rely on the capacity to approximate a random vector far beyond its closest lattice vector element (the Gap-CVP problem).

Traditionally, this approach was not developed for code-based cryptography since no decoding algorithm is known that decodes beyond the Gilbert-Varshamov bound: in fact this problem is somewhat marginal for the coding community since it implies many possibilities for decoding, while the emphasis is almost always to find the most probable codeword or a short list of most likely codewords.

Our contribution

The main contribution of this paper is the introduction of a new way of considering code-based signatures, by introducing the idea that it is possible to invert a random syndrome not below the Gilbert-Varshamov bound, but above it. The approach is similar in spirit to what is done in lattice-based cryptography. We describe a new algorithm for LRPC codes, a recently introduced class of rank codes, the new algorithm permits in practice decoding both errors and (generalized) rank erasures. This new algorithm enables us to approximate a syndrome beyond the Gilbert-Varshamov bound. The algorithm is a unique decoder (not a list decoder) but can give different solutions depending on the choice of the erasure. We explain precisely in which conditions one can obtain successful decoding for any given syndrome and give the related probabilistic analysis. Based on this error/erasure algorithm we propose a new signature scheme RankSign. We give conditions for which no information leakage is possible from real signatures obtained through our scheme. This point is very important since information leaking from real signatures was the weakness through which the NTRUSign scheme came to be attacked [19, 6, 26]. Finally, we give examples of parameters: they are rather versatile, and their size depends on a bound on the amount of potentially leaked information. In some cases one obtains public keys of size 5700 bits with signatures of length 1728 bits, moreover the scheme is rather fast.

The paper is organized as follows: Section 2 recalls basic facts on the rank metric, Section 3 introduces LRPC codes and describes a new mixed algorithm for decoding (generalized) erasures and errors, and studies its behaviour, Section 4 shows how to use them for cryptography, and lastly, Section 5 and 6 consider security and parameters for these schemes. The details of some proofs are also given in the appendix.

2 Background on rank metric codes and cryptography

2.1 Definitions and notation

Notation : Let q be a power of a prime p, m an integer and let V_n be a n dimensional vector space over the finite field $\mathrm{GF}(q^m)$. Let $\beta = (\beta_1, \ldots, \beta_m)$ be a basis of $GF(q^m)$ over $GF(q)$. Let $\mathcal{F}_i$ be the map from $GF(q^m)$ to $GF(q)$ where $\mathcal{F}_i(x)$ is the i-th coordinate of x in the basis β.

To any $v = (v_1, \ldots, v_n)$ in V_n we associate the matrix $\overline{v} \in \mathcal{M}_{m,n}(F_q)$ in which $\overline{v}_{i,j} = \mathcal{F}_i(v_j)$. The rank weight of a vector v can be defined as the rank of the associated matrix $\overline{v}$. If we name this value $\mathrm{rank}(v)$ we can define a distance between two vectors x, y through the formula $\mathrm{d_r}(x, y) = \mathrm{rank}(x - y)$. We refer to [21] for more details on codes for the rank distance.

A rank code C of length n and dimension k over $GF(q^m)$ is a subspace of dimension k of $GF(q^m)$ viewed as a (rank) metric space. The minimum rank distance of the code C is the minimum rank of non-zero vectors of the code. In the following, C is a rank metric code of length n and dimension k over $GF(q^m)$. The matrix G denotes a $k \times n$ generator matrix of $\mathcal{C}$ and H one of its parity check matrix.

Definition 1. *Let* $x = (x_1, x_2, \cdots, x_n) \in GF(q^m)^n$ *be a vector of rank* r. *We denote* E *the* $GF(q)$-*sub vector space of* $GF(q^m)$ *generated by* $x_1, x_2, \cdots, x_n$. *The vector space* E *is called the* **support** *of* x.

Remark 1. The notion of support of a code word for the Hamming distance and for the the one introduced in definition 1 are different but they share a common principle: in both cases, suppose one is given a syndrome s and that there exists a low weight vector x such that $H.x^t = s$, then, if the support of x is known, it is possible to recover all the coordinates values of x by solving a linear system.

Definition 2. *Let* e *be an error vector of rank* r *and error support space* E. *We denote by* **generalized erasure of dimension** t *of an error* e, *a subspace* T *of dimension* t *of its error support* E.

The notion of erasure for Hamming distance corresponds to knowing a particular position of the error vector (hence some partial information on the support), in the rank distance case, the support of the error being a subspace E, the equivalent notion of erasure (also denoted generalized erasure) is therefore the knowledge of a subspace T of the error support E.

2.2 Bounds for rank metric codes

The classical bounds for the Hamming metric have straightforward rank metric analogues, since two of them are of interest for the paper we recall them below.

Rank Gilbert-Varshamov bound [GVR] The number of elements $S(m, q, t)$ of a sphere of radius t in $GF(q^m)^n$, is equal to the number of $m \times n$ q-ary matrices of rank t. For $t = 0$

$S_0 = 1$, for $t \geq 1$ we have (see [21]):

$$S(n, m, q, t) = \prod_{j=0}^{t-1} \frac{(q^n - q^j)(q^m - q^j)}{q^t - q^j}$$

From this we deduce the volume of a ball $B(n, m, q, t)$ of radius t in $GF(q^m)$ to be:

$$B(n, m, q, t) = \sum_{i=0}^{t} S(n, m, q, i)$$

In the linear case the Rank Gilbert-Varshamov bound $GVR(n, k, m, q)$ for a $[n, k]$ linear code over $GF(q^m)$ is then defined as the smallest integer t such that $B(n, m, q, t) \geq q^{m(n-k)}$.

The Gilbert-Varshamov bound for a rank code C with dual matrix H, corresponds to the smallest rank weight r for which, for any syndrome s, there exists on the average a word x of rank weight r such that $H.x^t = s$. To give an idea of the behaviour of this bound, it can be shown that, asymptotically in the case $m = n$ ([21]): $\frac{GVR(n,k,m,q)}{n} \sim 1 - \sqrt{\frac{k}{n}}$.

Singleton bound The classical Singleton bound for a linear $[n, k]$ rank code of minimum rank r over $GF(q^m)$ works in the same way as for linear codes (by finding an information set) and reads $r \leq 1 + n - k$: in the case when $n > m$ this bound can be rewritten as $r \leq 1 + \lfloor \frac{(n-k)m}{n} \rfloor$ [21]. Codes achieving this bound are called Maximum Rank Distance codes (MRD).

2.3 Cryptography and rank codes

The main use of rank codes in the cryptographic context is through the rank analogue of the classical syndrome decoding problem.

Maximum Likelihood - Rank Syndrome Decoding problem (ML-RSD) Let H be a $(n - k) \times n$ matrix over $GF(q^m)$ with $k \leq n$, $s \in GF(q^m)^{n-k}$. The problem is to find the smallest weight r such that $\text{rank}(x) = r$ and $Hx^t = s$.

In that case it is not proven that the problem is NP-hard, but this problem is very close to the syndrome decoding problem which is NP-hard, moreover the problem can be seen as a structured version of the MinRank problem which is also NP-hard (the RSD problem can be attacked as a MinRank problem but in practice the attack does not work since there are too many unknowns [8]). Moreover the problem has been studied for more than 20 years and the best algorithms are exponential, so that the problem is generally believed to be hard.

There exist several types of generic attacks on the problem:

- **combinatorial attacks**: these attacks are usually the best ones for small values of q (typically $q = 2$) and when n and k are not too small (typically 30 and more), when q increases, the combinatorial aspect makes them less efficient. The first non-trivial attack on the problem was proposed by Chabaud and Stern [5] in 1996, then in 2002 Ourivski and Johannson [25] improved the previous attack and proposed a new attack, meanwhile these two attacks did

not take account of the value of n in the exponent. Very recently the two previous attacks were generalized in [15] by Gaborit et *al.* in $(n-k)^3 m^3 q^{(r-1)\lfloor \frac{(k+1)m}{n} \rfloor})$) and take the value of n into account and were used to break some repaired versions of the GPT cryposystem.

- **algebraic attacks**: the particular nature of the rank metric makes it a natural field for algebraic attacks and solving by Groebner basis, since these attack are largely independent of the value of q and in some cases may also be largely independent of m. These attacks are usually the most efficient when q increases and when the parameters are not too high (say less than 30). There exist different types of algebraic equations settings: the first one by Levy and Perret [20] in 2006 considers a quadratic setting by taking as unknowns the support E of the error and the error coordinates regarding E, there is also the Kernel attack by [8] and the minor approach which consists in considering multivariate equations of degree $r+1$ obtained from minors of matrices [9], and more recently the annulator setting by Gaborit et *al.* in [15] (which is valid on certain type of parameters but may not be independent on m). In our context for some of the parameters considered in the end of the paper, the Levy-Perret attack is the most efficient one to consider. All the complexities for Grobner basis attacks are estimated through the very nice program of L. Bettale [4].

The case of more than one solution: approximating beyond the GVR bound

In code based cryptography there is usually only one solution to the syndrome problem (for instance for the McEliece scheme), now in this situation we are interested in the case when there are a large number of solutions. This case is reminiscent of lattice-based cryptography when one tries to approximate as much as possible a given syndrome by a word of weight as low as possible.

This motivates us to introduce a new problem which corresponds to finding a solution to the general decoding problem for the case when the weight of the word associated to the syndrome is greater than the GVR bound, in that case there may be several solutions, and hence the term decoding does not seem well chosen. Notice that in a lattice cryptography context, it corresponds to the case of Gap-CVP, which does not make sense here, since it implies a multiplicative gap.

Approximate - Rank Syndrome Decoding problem (App-RSD) Let H be a $(n-k)\times n$ matrix over $GF(q^m)$ with $k \leq n$, $s \in GF(q^m)^{n-k}$ and let r be an integer. The problem is to find a solution of rank r such that $\text{rank}(x) = r$ and $Hx^t = s$.

It is help to first consider the situation of a binary linear $[n,k]$ Hamming metric code. Given a random element of length $n-k$ of the syndrome space, we know that with high probability there exists word that has this particular syndrome and whose weight is on the GV bound. This word is usually hard to find, however. Now what is the lowest minimum weight for which it is easy to find such a word ? A simple approach consists in taking $n-k$ random column of the parity-check matrix (a potential support of the solution word) and inverting the associated matrix, multiplying by the syndrome gives us a solution of weight $(n-k)/2$ on average. In fact it is difficult to do better than this without a super-polynomial increase in complexity.

Now for the rank metric, one can apply the same approach: suppose one starts from a random $[n,k]$ code over $GF(q^m)$ and that one searches for a word of small rank weight r with a given syndrome. One fixes (as in the Hamming case) a potential support for the word - here a subspace of dimension r of $GF(q^m)$- and one tries to find a solution. Let $x = (x_1, \cdots, x_n)$

be a solution vector, so that $H.x^t = s$. If we consider the syndrome equations induced in the small field $GF(q)$, there are $n.r$ unknowns and $m.(n-k)$ equations. Hence it is possible (with a good probability) to solve the system whenever $nr \geq m(n-k)$, therefore it is possible to find in probabilistic polynomial time a solution to a typical instance of the RSD problem whenever $r \geq \frac{m(n-k)}{n} = 1 + \lfloor \frac{m(n-k)}{n} \rfloor$, which corresponds to the Singleton bound (except if $\frac{m(n-k)}{n}$ is an integer).

This proves the following proposition:

Proposition 1. *There is a probabilistic polynomial time algorithm that solves random instances of the App-RSD problem in polynomial time when $r \geq 1 + \lfloor \frac{m(n-k)}{n} \rfloor$ or for $r \geq \frac{m(n-k)}{n}$ if $\frac{m(n-k)}{n}$ is an integer.*

For a rank weight r below this bound, the best known attacks are, as in the Hamming distance case, obtained by considering the cost of finding a word of rank r divided by the number of potential solutions: $\frac{B(n,k,m,q)}{q^{m(n-k)}}$. In practice the complexity we find is coherent with this.

3 Approximating a random syndrome beyond the GVR bound with LRPC codes

3.1 Decoding algorithm in rank metric

The rank metric has received a lot of attention in the context of network coding [27]. There exist very few algorithms, however, for decoding codes in the rank metric. The most well known $[n, k]$ codes which are decodable are the Gabidulin codes [11]. These codes can correct up to $\frac{n-k}{2}$ errors, and have been proposed for encryption: but since they cannot decode up to the GVR bound, they do not seem suitable for full decoding in the spirit of [7] for signature algorithms. Another more recent family of decodable codes are the LRPC codes [13], these codes are defined through a low rank matrix.

Definition 3. *A Low Rank Parity Check (LRPC) code of rank d, length n and dimension k over $GF(q^m)$ is a code defined by an $(n - k) \times n$ parity check matrix $H = (h_{ij})$, such that all its coordinates h_{ij} belong to the same $GF(q)$-subspace F of dimension d of $GF(q^m)$. We denote by $\{F_1, F_2, \cdots, F_d\}$ a basis of F.*

These codes can decode with a good probability up to $\frac{n-k}{d}$ errors, they can be used for encryption [13], but since they can decode only up to $\frac{n-k}{2}$ errors at best, they also seems unsuitable for signature algorithms.

3.2 Using LRPC codes to approximate a random syndrome beyond the GVR bound

In this section we explain how LRPC codes can be used to approximate a random syndrome beyond the GVR bound.

High level overview The traditional approach for decoding random syndromes, that is used by the CFS scheme for instance, consists in taking advantage of the decoding properties of a code (e.g. a Goppa code) and considering parameters for which the proportion of decodable vectors – the decodable density – is not too low. For the Hamming metric, this approach leads to very flat dual matrices, ie, codes with high rate and very low Hamming distance. In the rank metric case, this approach leads to very small decodable densities and does not work in practice. However, it is possible to proceed otherwise. It turns out that the decoding algorithm of LRPC codes can be adapted so that it is possible to decode not only errors but also (generalized) erasures. This new decoding algorithm allows us to decode more rank errors since the support is then partially known. In that case since the size of the balls depends directly on the dimension of the support, it leads to a dramatic increase of the size of the decodable balls. Semantically, what happens is that the signer can fix an erasure space, which relaxes the condition for finding a preimage. This approach works because in the particular case of our algorithm, it is possible to consider the erasure space at no cost in terms of error correction: to put it differently, the situation for LRPC is different from traditional Hamming metric codes for which "an error equals two erasures".

In practice it is possible to find parameters (not flat at all) for which it is possible to decode a random syndrome with the constraint that its support contains a fixed random subspace. Fixing part of the rank-support of the error, (the generalized erasure) allows us more rank-errors. For suitable parameters, the approach works then as follows: for a given random syndrome-space element s, one chooses a random subspace T of fixed dimension t (a generalized erasure of Definition 2), and the algorithm returns a small rank-weight word, whose rank-support E contains T, and whose syndrome is the given element s. Of course, there is no unicity of the error e since different choices of T lead to different errors e, which implies that the rank of the returned error is above the GVR bound: it is however only just above the GVR bound for the right choice of parameters.

LRPC decoding with errors and generalized erasures

Setting: Let an $[n, k]$ LRPC code be defined by an $(n - k) \times n$ parity-check matrix H whose entries lie in a space $F \subset GF(q^m)$ of small dimension d. Let t and r' be two parameters such that

$$r' \leq \frac{n - k}{d}.$$

Set $r = t + r'$. Given an element of the syndrome space s, we will be looking for a rank r vector e of $GF(q^m)^n$ with syndrome s. We first look for an acceptable subspace E of dimension r of $GF(q^m)$ and then solve the linear system $H.e^t = s$ where $e \in E^n$. To this end we choose a random subspace T of dimension t of $GF(q^m)$ and impose the condition $T \subset E$.

The subspace T being fixed, we now describe the set of decodable elements of the syndrome space. We will then see how to decode them.

Definition 4. *Let F_1 and F_2 be two fixed linearly independent elements of the space F. We shall say that an element $s \in GF(q^m)^{n-k}$ of the syndrome space is T-decodable if there exists a rank r subspace E of $GF(q^m)$ satisfying the following conditions.*

(i) $\dim \langle FE \rangle = \dim F \dim E$,

(ii) $\dim(F_1^{-1}\langle FE \rangle \cap F_2^{-1}\langle FE \rangle) = \dim E$,

(iii) *the coordinates of s all belong to the space $\langle FE \rangle$ and together with the elements of the space $\langle FT \rangle$ they generate the whole of $\langle FE \rangle$.*

Decoding algorithm. We now argue that if a syndrome s is T-decodable, we can effectively find e of rank r such that $H.e^t = s$. We first determine the required support space E. Since the decoder knows the subspaces F and T, he has access to the product space $\langle FT \rangle$. He can then construct the subspace S generated by $\langle FT \rangle$ and the coordinates of s. Condition (iii) of T-decodability ensures that the subspace S is equal to $\langle FE \rangle$ for some E, and since

$$F_1^{-1}\langle FE \rangle \cap F_2^{-1}\langle FE \rangle \supset E,$$

condition (ii) implies that E is uniquely determined and that the decoder recovers E by computing the intersection of subspaces $F_1^{-1}S \cap F_2^{-1}S$.

It remains to justify that once the subspace E is found, we can always find e of support E such that $H.e^t = s$. This will be the case if the mapping

$$\begin{aligned} E^n &\to \langle FE \rangle^{n-k} \\ e &\mapsto H.e^t \end{aligned} \tag{1}$$

can be shown to be surjective. Extend $\{F_1, F_2\}$ to a basis $\{F_1, \cdots, F_d\}$ of F and let $\{E_1, \cdots, E_r\}$ be a basis of E. Notice that the system $H.e^t = s$ can be rewritten formally as a linear system in the small field $GF(q)$ where the coordinates of e and the elements of H are written in the basis $\{E_1, \cdots, E_r\}$ and $\{F_1, \cdots, F_d\}$ respectively, and where the syndrome coordinates are written in the product basis $\{E_1.F_1, \cdots, E_r.F_d\}$. We therefore have a linear system with $n.r$ unknowns and $(n-k).rd$ equations over $GF(q)$ that is defined by an $(n.r) \times (n-k)rd$ formal matrix H_f (*say*) whose coordinates are functions only of H (see [13] for more details on how to obtain H_f from H).

We now see that the matrix H can be easily chosen so that the matrix H_f is of maximal rank $n.r$, which makes the mapping (1) surjective, *for any subspace E of dimension d satisfying* condition (i) of T-decodability.

Remarks:

1. For applications, we will consider only the case where $nr = (n-k)rd$, meaning that the mapping (1) is always one-to-one.
2. The system $H.e^t = s$ can be formally inverted and stored in a pre-processing phase, so that the decoding complexity is only that of multiplication by a square matrix of length nr, rather than a cubic inversion.
3. In principle, the decoder could derive the support E by computing

$$E = F_1^{-1}S \cap \cdots \cap F_d^{-1}S \tag{2}$$

   rather than simply $E = F_1^{-1}S \cap F_2^{-1}S$, and the procedure would work in the same way in cases when (2) holds but not the simpler condition (ii). This potentially increases the set of decodable syndromes, but the gain is somewhat marginal and condition (ii) makes the forthcoming analysis simpler. For similar reasons, when conditions (i)–(iii) are not all satisfied, we do not attempt to decode even if there are cases when it stays feasable.

Figure 1 summarizes the decoding algorithm. Note that the decoder can easily check conditions (i)–(iii), and that a decoding failure is declared when they are not satisfied.

Input: $T = \langle T_1, \cdots, T_t \rangle$ a subspace of $GF(q^m)$ of dimension t, H an $(n-k) \times n$ matrix with elements in a subspace $F = \langle F_1, \cdots, F_d \rangle$ of dimension d, and $s \in GF(q^m)^{n-k}$.
Output: a vector $e = (e_1, \ldots e_n)$ such that $s = H.e^t$, with $e_i \in E$, E a subspace of dimension $\dim E = r = t + \frac{n-k}{d}$ satisfying $T \subset E$.

1. **Syndrome computations**
   a) Compute a basis $B = \{F_1 T_1, \cdots, F_d T_t\}$ of the product space $\langle F.T \rangle$.
   b) Compute the subspace $S = \langle B \cup \{s_1, \cdots, s_{n-k}\} \rangle$.

2. **Recovering the support E of the error**
   Compute the support of the error $E = F_1^{-1} S \cap F_2^{-1} S$, and compute a basis $\{E_1, E_2, \cdots, E_r\}$ of E.

3. **Recovering the error vector $e = (e_1, \ldots, e_n)$**
   For $1 \le i \le n$, write $e_i = \sum_{i=1}^{n} e_{ij} E_j$, solve the system $H.e^t = s$, where the equations $H.e^t$ and the syndrome coordinates s_i are written as elements of the product space $P = \langle E.F \rangle$ in the basis $\{F_1 E_1, \cdots, F_1 E_r, \cdots, F_d E_1, \cdots, F_d E_r\}$. The system has nr unknowns (the e_{ij}) in $GF(q)$ and $(n-k).rd$ equations from the syndrome.

Fig. 1. Algorithm 1: a general errors/erasures decoding algorithm for LRPC codes

3.3 Proportion of decodable syndromes for unique decoding of LRPC codes

Signature algorithms based on codes all inject the message space in some way into the syndrome space and then decode them to form a signature. We should therefore estimate the proportion of decodable syndromes. The classical decoding approach tells us to look for a preimage by H that sits on the Gilbert-Varshamov bound: for typical random codes, a preimage typically exists and is (almost) unique. Computing such a preimage is a challenge, however. In our case, we are looking for a preimage above the Gilbert-Varshamov bound, for which many preimages exist, but for a fixed (erasure) subspace T, decoding becomes unique again. In the following, we count the number of T-decodable syndromes and show that for some adequate parameter choices, their proportion can be made to be close to 1.

It will be convenient to use the following notation.

Definition 5. *For a subspace T of $GF(q^m)$ of dimension t, denote by $\mathcal{E}(T)$ the number of subspaces of dimension $r = r' + t$ that contain T.*

Lemma 1. *We have*

$$\mathcal{E}(T) = \prod_{i=0}^{r'-1} \left(\frac{q^{m-t-i} - 1}{q^{i+1} - 1} \right)$$

Proof. Consider the case where $r = t+1$, we need to construct distinct subspaces of dimension $t+1$ containing T. This can be done by adjoining an element of $GF(q^m)$ modulo the subspace T, which gives $(q^m - q^t)/(q^{t+1} - q^t) = (q^{m-t} - 1)/(q - 1)$ possibilities. Now any subspace of dimension $t+1$ contains $q^{t+1} - 1$ supspaces of dimension t containing T. A repetition of this approach $r' - 1$ times gives the formula. (see also [22] p.630). □

Theorem 1. *The number $\mathcal{T}(t, r, d, m)$ of T-decodable syndromes satisfies the upper bound:*

$$\mathcal{T}(t, r, d, m) \le \mathcal{E}(T) q^{rd(n-k)}.$$

9

Furthermore, under the conditions $r(2d-1) \leq m$ and

$$\dim\langle FT \rangle = \dim F \dim T, \tag{3}$$

$$\dim(F_1^{-1}F + F_2^{-1}F) = 2\dim F - 1 = 2d - 1, \tag{4}$$

we also have the lower bound:

$$\left(1 - \frac{1}{q-1}\right)^2 \mathcal{E}(T)q^{rd(n-k)} \leq \mathcal{T}(t,r,d,m).$$

Note that condition (4) depends only on the subspace F and can be ensured quite easily when designing the matrix H. Random spaces F with random elements F_1 and F_2 will typically have this property. Condition (3) depends on the choice of the subspace T: as will be apparent from Lemma below, for a random subspace T condition (3) holds with probability very close to 1.

To prove Theorem 1 we will rely on the following lemma, whose proof we relegate to the Appendix.

Lemma 2. *Let A be a fixed subspace of $\mathbb{F}_q^m$ of dimension α and let T be a subspace of dimension t (with possibly $t = 0$) such that $\dim\langle AT \rangle = \alpha t$. Let B be a subspace generated by T together with β random independent uniform vectors, with β satisfying $\alpha(t+\beta) \leq m$. Then*

$$\mathbb{P}\left(\dim\langle AB \rangle < \alpha(t+\beta)\right) \leq \frac{q^{\alpha(t+\beta)}}{(q-1)q^m}.$$

Proof of Theorem 1. To obtain a T-decodable syndrome, we must choose $n - k$ elements in a space $\langle FE \rangle$ for a given space E that contains T. There are $\mathcal{E}(T)$ ways of choosing E, and for any given E there are at most $q^{\dim F \dim E} = q^{dr}$ ways of choosing a syndrome coordinate in $\langle FE \rangle$. This gives the upper bound on $\mathcal{T}(t,r,d,m)$.

We proceed to prove the lower bound. First consider that Lemma 2 proves that, when we randomly and uniformly choose a subspace E that contains T, then with probability at least $1 - 1/(q-1)$, we have:

$$\dim\langle (F_1^{-1}F + F_2^{-1}F)E \rangle = \dim(F_1^{-1}F + F_2^{-1}F)\dim E = (2d-1)r$$

by property (4). This last fact implies, that

$$\dim(F_1^{-1}\langle FE \rangle + F_2^{-1}\langle FE \rangle) = 2dr - r \tag{5}$$

since clearly

$$F_1^{-1}\langle FE \rangle + F_2^{-1}\langle FE \rangle = \langle (F_1^{-1}F + F_2^{-1}F)E \rangle.$$

Now, since we have $E \subset F_1^{-1}F \cap F_2^{-1}F$, applying the formula $\dim(A+B) = \dim A + \dim B - \dim A \cap B$ to (5) gives us simultaneously that:

$$\dim\langle FE \rangle = dr$$

$$F_1^{-1}F \cap F_2^{-1}F = E.$$

In other words, both conditions (i) and (ii) of T-decodability are satisfied. We have therefore proved that the proportion of subspaces E containing T that satisfy conditions (i) and (ii) is at least $(1 - 1/(q - 1))$.

Now let E be a fixed subspace satisfying conditions (i) and (ii). Among all $(n - k)$-tuples of elements of $\langle FE \rangle$, the proportion of those $(n - k)$-tuples that together with $\langle FT \rangle$ generate the whole of $\langle FE \rangle$ is at least

$$\left(1 - \frac{1}{q}\right)\left(1 - \frac{1}{q^2}\right) \cdots \left(1 - \frac{1}{q^i}\right) \cdots \geq 1 - \frac{1}{q - 1}. \tag{6}$$

We have therefore just proved that given a subspace E satisfying conditions (i) and (ii), there are at least $(1 - 1/q)(q^{rd})^{n-k}$ $(n - k)$-tuples of $\langle FE \rangle^{n-k}$ satisfying condition (iii).

To conclude, notice that since a T-decodable syndrome entirely determines the associated subspace E, the set of T-decodable syndromes can be partitioned into sets of $(n - k)$-tuples of $\langle FE \rangle^{n-k}$ satisfying condition (iii) for all E satisfying conditions (i) and (ii). The two lower bounds on the number of such E and the number of T-decodable syndromes inside a given $\langle FE \rangle^{n-k}$ give the global lower bound of the Theorem. □

Remarks:

1. It can be shown with a finer analysis that the term $(1 - 1(q - 1))^2$ in the lower bound can be improved to a quantity close to $1 - 1(q - 1)$.
2. For large q, Theorem 1 shows that, for most choices of T, the density of T-decodable syndromes essentially equals

$$\frac{\mathcal{E}(T)q^{rd(n-k)}}{q^{m(n-k)}} \approx q^{(r-t)(m-r)+(n-k)(rd-m)}. \tag{7}$$

Remarkably, it is possible to choose sets of parameters (m, t, r, d), with $(n - k) = d(r - t)$, such that the exponent in (7) equals zero, which gives a density very close to 1.

Example of parameters with density almost 1: $q = 2, m = 45, n = 40, k = 20, t = 5, r = 10$, the algorithm decodes up to $r = t + r' = 15$ for a fixed random partial support T of dimension 5. The GVR bound for a random $[40, 20]$ code with $m = 45$ is 13, the Singleton bound is 20, we see that the decoding radius 15 is therefore just above the GVR bound at 13 and rather far from the Singleton bound at 20. Moreover one can notice that if parameters (m, t, r, d) satisfy the two equations $(r - t)(m - r) + (n - k)(rd - m) = 0$ and $(n - k) = d(r - t)$ (the case for which the density is almost 1), then for any α integer greater than 1, the set $(\alpha m, \alpha t, \alpha r, d)$ satisfy the same equations, and hence for a given d one obtains an infinite family of parameters with density almost 1.

Decoding in practice. In practice it easy enough to find sets of parameters for which the density of decodable syndromes is very close to 1, i.e. such that $(r - t)(m - r) + (n - k)(rd - m) = 0$. When $q = 2$, the formula (7) is only an approximation, and simulations give a density close to $1/3$ rather than 1.

4   RankSign : a rank-based signature scheme

We saw in the previous section how to construct a matrix H of an LRPC code, with a unique support decoding, which opens the way for a signature algorithm. In practice the best

decoding results are obtained for $d = 2$: the natural strategy is to define for the public key a matrix $H' = AHP$, where A is a random $(n-k) \times (n-k)$ invertible matrix in the extension field and P is an invertible $n \times n$ matrix in the small field. However, it is easily possible for a cryptanalyst to recover the words of small weight $d = 2$ in H' and it is therefore necessary to hide the matrix H in another way. In what follows we present a simple type of masking: $RankSign^+$ which consists in adding a few random columns to H, other more complex types of masking are also possible: $RankSign^\times$ (RankSign-multiply) and $RankSign^{+\times}$, which are presented in Appendix C.

4.1 The $RankSign^+$ scheme: augmented LRPC codes

Suppose one has a fixed support T of dimension t. We consider the public matrix $H' = A(R|H)P$ with R a random $(n-k) \times t'$ matrix in $GF(q^m)$. We will typically take $t' = t$ but one could envisage other values of t'. We denote by *augmented LRPC codes* such codes with parity-check matrices $H' = A(R|H)P$.

Starting from a partial support T that has been randomly chosen and is then fixed, the signature consists in decoding not a random s but the syndrome $s' = s - R.(e_1, \cdots, e_t)^t$ for e_i random independent elements of T.

The overall rank of the solution vector e is still $r = t + r'$. the masking gives us that the minimum rank-weight of the code generated by the rows of H' is $t + d$ rather than purely d: therefore recovering the hidden structure involves finding relatively large minimum weight vectors in a code. In practice we consider $d = 2$ and H is a $n/2 \times n$ matrix with all coordinates in a space F of dimension 2. Moreover for $\{F_1, F_2\}$ a basis of F, we choose the matrix H such that when H is written in the basis $\{F_1, F_2\}$, one obtains a $n \times n$ invertible matrix (of maximal rank) over $GF(q)$. It can be done easily. Figure 2 describes the scheme, where $||$ denotes concatenation.

---

1. **Secret key:** an augmented LRPC code over $GF(q^m)$ with parity-check matrix $(R|H)$ of size $(n-k) \times (n+t)$ which can decode r' errors and t generalized erasures: a randomly chosen $(n-k) \times (n-k)$ matrix A that is invertible in $GF(q^m)$ a randomly chosen $(n+t) \times (n+t)$ matrix P invertible in $GF(q)$.
2. **Public key:** the matrix $H' = A(R|H)P$, a small integer value l, a hash function *hash*.
3. **Signature of a message M:**
   a) *initialization*: seed $\leftarrow \{0,1\}^l$, pick t random independent elements $(e_1, \cdots, e_t)$ of $GF(q^m)$
   b) *syndrome*: $s \leftarrow hash(M||seed) \in GF(q^m)^{n-k}$
   c) decode by the LRPC matrix H, the syndrome $s' = A^{-1}.s^T - R.(e_1, \cdots, e_t)^T$ with erasure space $T = \langle e_1, \cdots, e_t \rangle$ and r' errors by Algorithm 1.
   d) if the decoding algorithm works and returns a word $(e_{t+1}, \cdots, e_{n+t})$ of weight $r = t + r'$, signature$=((e_1, \cdots, e_{n+t}).(P^T)^{-1}, seed)$, else return to a).
4. **Verification**: Verify that $Rank(e) = r = t + r'$ and $H'.e^T = s = hash(M||seed)$.

Fig. 2. The $RankSign^+$ signature algorithm

Parameters: *Public key size:* $(k+t)(n-k)mLog_2(q)$ *Signature size:* $(m+n+t)rLog_2(q)$.

The cost of the decoding algorithm is quadratic because of preprocessing of H_f^{-1}, hence the major cost comes from the linear algebra over the large field $GF(q^m)$.

Signature complexity: $(n - k) \times (n + t)$ operations in $GF(q^m)$. *Verification complexity:* $(n - k) \times (n + t)$ operations in $GF(q^m)$.

The length l of the seed can be taken as $\frac{80}{Log_2(q)}$ for instance.

4.2   Optimized parameters: cyclic-RankSign

In order to decrease the size of the public key it is possible to use a quasi-cyclic structure. Whenever all parameters n, k, t or/and l are divisible by a certain cyclicity order σ it is possible to consider that all matrices consist of small cyclic $\sigma \times \sigma$ random matrices. This particular type of structure divides the size of the public by a factor σ.

5   Security

5.1   A difficult problem

We now introduce a problem to which the security of our scheme can be reduced, we then discuss the difficulty of the problem.

Approximate Syndrome Decoding Problem for augmented LRPC codes (App-RSD-LRPC-Aug)

Given a masked parity-check matrix $H' = A(R|H)P$ of an augmented LRPC codes as defined in the previous section, and a random syndrome s, find a vector x of not too small rank d (typically of rank $(GVR + Singleton)/2$) such that $H'.x^t = s$.

Assumption: *The problem App-RSD-LRPC-Aug is difficult .*

Discussion on the assumption: The family of augmented LRPC codes is not of course a family of random codes, but they are weakly structured codes: the main point being that they have a parity-check matrix one part of which consists only in low rank coordinates the other part consisting in random entries. The attacker never has direct access to the LRPC matrix H, which is hidden by the augmented part.

There are two different types of attacks on the problem: structural attacks in which the attacker tries to recover the structure of the code, or the attacker can try to forge directly a signature and try to find codewords of weight $r = t + r'$. We consider in the following these two approaches and explain why it is difficult to attack this problem.

Remark There is a long history of broken cryptosystems related to hiding structured matrices. One can though remark the following: 1) there are strongly structured families of hidden codes like the Goppa-McEliece scheme which have not been attacked, 2) the families of hidden structures which have been attacked rely usually on **very** strongly structured codes like the Reed-Solomon codes or the Gabidulin codes [10], which are very difficult to hide. Moreover there exist hidden families of codes or lattices with a low structure which have never been really attacked like for instance the NTRU lattices or the double circulant codes. It seems that the weak structure of the LRPC codes place them in this latter category.

Structural attacks on the public key: trying to recover the secret key

Previously known structural attacks The main structural attack for the rank metric is the Overbeck attack on the GPT cryptosystem, the attack consists in considering concatenated public matrices $G^q, G^{q^2}, ..., G^{q^{n-k-}}$, in that case the particular structure of Gabidulin codes enables one to find a concatenated matrix with a rank default; this is due to the particular structure of the Gabidulin codes and the fact that for Gabidulin codes G^{q^i} is very close to $G^{q^{i+1}}$. In the case of LRPC codes, since the rows are taken randomly in a small space, this attack makes no sense, and cannot be generalized.

Attack on the parity-check matrix H': another approach consists in finding directly words of small weight induced by the structure of the code, from which one can hope to recover the global structure. For augmented LRPC codes, the rank of the minimum weight words is $d+t$: d for LRPC and t for the masking. This attack becomes very hard when t increases, even for low t. For instance for $t = 2$ and $d = 2$ it gives a minimum weight of 4, which for most parameters n and k is already out of reach of the best known attacks on the rank syndrome decoding (see Section 2).

Attack on the isometry matrix P: The attacker can also try to guess the action of P on H, since d is usually small negating this action may permit to attack directly a code of rank d. Since d is small it is enough to guess the resulting action of P on $n - k + 3$ columns by considering only the action of P coming from the first t columns of the matrix R - the only columns which may increase the rank-, it means guessing $(n - k + 3) \times t$ elements of $GF(q)$ (since coordinates of P are in $GF(q)$, hence a complexity of $q^{(n-k+3)t}$. In general this attack is not efficient, except in the case of quasi-cyclic matrices and small q, in which case the complexity becomes $q^{\frac{(n-k+3)t}{\sigma}}$, for σ the index of quasi-cyclicity.

Direct attack on the syndrome In that case one has to find a word of weight between the GVR bound and the Singleton bound, as discussed in Section 2 the best attack consists in computing the complexity of finding a word of given weight divided by the number of such words (the traditionnal approach for Hamming metric, which is also applicable here). Parameters are chosen in order to resist this attack.

Information leakage. We considered in previous attacks the case where no additional information was known besides the public parameters. Often the most efficient attacks on signatures is to recover the hidden structure of the public key by using information leaking from real signatures. This for instance is what happened in the case of NTRUSign: the secret key is not directly attacked, but the information leaked from real signatures enables one to recover successfully the hidden structure. We show in the next section that with our masking scheme no such phenomenon can occur, since we prove that, if an attacker can break the signature scheme for public augmented matrices with the help of information leaking from a number of (approximately) q real signatures, then he can also break the scheme just as efficiently *without* any authentic signatures.

5.2 Unforgeability

Theorem 2 below states the unforgeability of signatures. It essentially states that valid signatures leak no information on the secret key. More precisely, under the random oracle model, there exists a polynomial time algorithm that takes as input the public matrix H' and, produces couples (m, σ), where m is a message and σ a valid signature for m when one's only access to the hashing oracle is through the simulator, and this with the same probability distribution as those output by the authentic signature algorithm. Therefore whatever forgery can be achieved from the knowledge of H' and a list of valid signed messages, can be simulated and reproduced with the public matrix H' as only input.

Theorem 2. : *For any algorithm $\mathcal{A}$ that leads to a forged signature using $N \leq q/2$ authentic signatures, there is an algorithm $\mathcal{A}'$ with the same complexity that leads to a forgery using only the public key as input and without any authentic signatures.*

Proof. see Appendix B □

Corollary 1. *Let us take $q > 2^{t_{sec}}$, with t_{sec} a security parameter polynomially equivalent to n. If a polynomial time forger succeeds in forging the signature with some probability, he can solve the difficult problem App-RSD-LRPC-Aug with the same complexity and the same probability.*

Proof. Theorem 2 tells us that either the forger used $N > q/2$ signatures, or he did not use any. In the case when the forger did not use signatures, the problem of forging a new signature is exactly the same as the problem App-RSD-LRPC-Aug. In the other case, the forger used $N > q/2$ signatures and the forger could not run in polynomial time in the parameter n. □

6 Practical security and parameters

6.1 Parameters

In the following we give some examples of parameters. The parameters are adjusted to resist all previously known attacks. The security reduction holds for up to $q/2$ signatures, hence if one considers $q = 2^{40}$ it means we are protected against leakage for up to 2^{40} obtained authentic signatures. such an amount of signatures is very difficult to obtain in real life, moreover if one multiplies by the amount of time necessary to obtain a signature (about 2^{30} for $q = 2^{40}$) we clearly see that obtaining such a number of authentic signatures is out of reach, and it justifies our security reduction.

We also give parameters for q lower than 2^{40}: in that case the reduction is weaker in the sense that it does not exclude a leaking attack for sufficiently many signatures. However, such a leaking attack seems difficult to obtain anyway, and these parameters can be seen as challenges for our system.

In the case of large q, the best attacks are structural attacks for recovering a vector of weight $t+d$ orthogonal to the matrix H'. The best attacks are algebraic attacks from Levy-Perret: for

example the case $n = 16$ gives 270 quadratic equations for 126 unknowns, with a theoretical complexity of 2^{120} from [4]. For smaller q, the best attacks are combinatorial (Gaborit *et al.*).

• $RankSign^+$

In that case the matrix H' is a $(n-k) \times (n+t)$ matrix.

n	n-k	m	q	t	r'	r	GVR	Singleton	public key (bits)	sign. size (bits)	security
16	8	18	2^{40}	2	4	6	5	8	57600	8640	130
16	8	18	2^8	2	4	6	5	8	11520	1728	120
16	8	18	2^{16}	2	4	6	5	8	23040	3456	120
32	16	39	2	5	8	13	10	16	13104	988	80
40	20	45	2	5	10	15	12	20	22500	1350	110

• $Cyclic - RankSign^+$

In that case the matrix H' is a $(n-k) \times (n+t)$ quasi-cyclic matrix, and all matrices A, P, R are formed with concatenation of $\sigma \times \sigma$ cyclic blocks. We consider solely the case $\sigma = 2$. The complexity may be a little smaller than in the previous case since in that case attacks that try to guess a part of P are more efficient because of the cyclicity.

n	n-k	m	q	t	r'	r	σ	Singleton	public key (bits)	sign. size (bits)	security
16	8	18	2^{40}	2	4	6	2	8	28300	8640	130
16	8	18	2^8	2	4	6	2	8	5760	1728	90
16	8	18	2^{16}	2	4	6	2	8	11520	3456	120

6.2   Implementation

We implemented our scheme in a non optimized way, the results we obtained showed that for small q the scheme was very fast, when q increases, one has to consider the cost of multiplication in $GF(q)$, however for $q = 2^8$ or $q = 2^{16}$ some optimized implementation may reduce this cost.

7   Conclusion

In this paper we introduced a new approach to devising signatures with coding theory and in particular in the rank metric, by proposing to decode both erasures and errors rather than simply errors. This approach enables one to return a small weight word beyond the Gilbert-Varshamov bound rather than below. We proposed a new efficient algorithm for decoding LRPC codes which makes this approach a realizable. We then proposed a signature scheme based on this algorithm and the full decoding of a random syndrome beyond the Gilbert-Varshamov bound. We also showed that it was possible to protect our system against leakage from authentic signatures. Overall we propose different types of parameters, some of which are rather small.

References

1. Thierry P. Berger, Pierre-Louis Cayrel, Philippe Gaborit, Ayoub Otmani: Reducing Key Length of the McEliece Cryptosystem. AFRICACRYPT 2009: 77-97
2. Thierry P. Berger, Pierre Loidreau: Designing an Efficient and Secure Public-Key Cryptosystem Based on Reducible Rank Codes. INDOCRYPT 2004: 218-229
3. Luk Bettale, Jean-Charles Faugère, Ludovic Perret: Hybrid approach for solving multivariate systems over finite fields. J. Mathematical Cryptology 3(3): 177-197 (2009)
4. http://www-polsys.lip6.fr/ bettale/hybrid
5. Florent Chabaud, Jacques Stern: The Cryptographic Security of the Syndrome Decoding Problem for Rank Distance Codes. ASIACRYPT 1996: 368-381
6. Léo Ducas, Phong Q. Nguyen: Learning a Zonotope and More: Cryptanalysis of NTRUSign Counter-measures. ASIACRYPT 2012: 433-450
7. Courtois N., Finiasz M. and Sendrier N. : How to achieve a McEliece based digital signature scheme. Proc. of Asiacrypt 2001, Springer LNCS Vol. 2248, pp. 157–174 (2001)
8. J.-C. Faugère, F. Levy-dit-Vehel, L. Perret. Cryptanalysis of MinRank. In *CRYPTO 2008*, LNCS 5157, pages 280–296. Springer Verlag, 2008.
9. Jean-Charles Faugère, Mohab Safey El Din, Pierre-Jean Spaenlehauer: Computing loci of rank defects of linear matrices using Gröbner bases and applications to cryptology. ISSAC 2010: 257-264
10. Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, Jean-Pierre Tillich: Algebraic Cryptanalysis of McEliece Variants with Compact Keys. EUROCRYPT 2010: 279-298
11. Ernst M. Gabidulin, Theory of Codes with Maximum Rank Distance, Probl. Peredachi Inf, (21), pp. 3-16 (1985).
12. Ernst M. Gabidulin, A. V. Paramonov, O. V. Tretjakov: Ideals over a Non-Commutative Ring and thier Applications in Cryptology. EUROCRYPT 1991: 482-489
13. P. Gaborit and G. Murat and O. Ruatta and G. Zémor, Low Rank Parity Check Codes and their application in cryptography. Published in Workshop Codes and Cryptography (WCC 2013), Bergen (available at http://www.selmer.uib.no/WCC2013/pdfs/Gaborit.pdf)
14. Philippe Gaborit, Julien Schrek, Gilles Zémor: Full Cryptanalysis of the Chen Identification Protocol. PQCrypto 2011: 35-50
15. P. Gaborit, O. Ruatta and J. Schrek, On the complexity of the rank syndrome decoding problem, eprint,http://arxiv.org/abs/1301.1026
16. Craig Gentry, Chris Peikert, Vinod Vaikuntanathan: Trapdoors for hard lattices and new cryptographic constructions. STOC 2008: 197-206
17. Oded Goldreich, Shafi Goldwasser, Shai Halevi: Public-Key Cryptosystems from Lattice Reduction Problems. CRYPTO 1997: 112-131
18. Jeffrey Hoffstein, Jill Pipher, Joseph H. Silverman: NTRU: A Ring-Based Public Key Cryptosystem. ANTS 1998: 267-288
19. Jeffrey Hoffstein, Nick Howgrave-Graham, Jill Pipher, Joseph H. Silverman, William Whyte: NTRUSIGN: Digital Signatures Using the NTRU Lattice. CT-RSA 2003: 122-140
20. F. Levy-dit-Vehel and L. Perret, Algebraic decoding of rank metric codes, proceedings of YACC06.
21. P. Loidreau, Properties of codes in rank metric, http://arxiv.org/abs/cs/0610057
22. J. MacWilliams and N.J.A. Sloane, "The theory of error correcting codes", North Holland, Ninth impression (1977)
23. Daniele Micciancio, Oded Regev, Lattice-based Cryptography Book chapter in Post-quantum Cryptography, D. J. Bernstein and J. Buchmann (eds.), Springer (2008
24. Rafael Misoczki and Jean-Pierre Tillich and Nicolas Sendrier and Paulo S. L. M. Barreto, MDPC-McEliece: New McEliece Variants from Moderate Density Parity-Check Codes Cryptology ePrint Archive: Report 2012/409
25. Ourivski, A. V. and Johansson, T., New Technique for Decoding Codes in the Rank Metric and Its Cryptography Applications,Probl. Inf. Transm.(38), 237–246 (2002)
26. Phong Q. Nguyen, Oded Regev: Learning a Parallelepiped: Cryptanalysis of GGH and NTRU Signatures. EUROCRYPT 2006: 271-288
27. D. Silva, Kschishang, R. Kötter, "Communication over Finite-Field Matrix Channels," IEEE Trans. Inf. Theory, vol. 56, pp. 1296–1305, Mar. 2010.
28. Stern J. : A new paradigm for public key identification. IEEE Transactions on Information Theory, IT 42(6), pp. 2757–2768 (1996)

A Typical dimensions for products of subspaces

We provide here a proof of Lemma 2 that we recall.

Lemma 2. *Let A be a fixed subspace of $\mathbb{F}_q^m$ of dimension α and let T be a subspace of dimension t (with possibly $t = 0$) such that $\dim\langle AT\rangle = \alpha t$. Let B be a subspace generated by T together with β random independent uniform vectors, with β satisfying $\alpha(t+\beta) \le m$. Then*

$$\mathbb{P}\left(\dim\langle AB\rangle < \alpha(t+\beta)\right) \le \frac{q^{\alpha(t+\beta)}}{(q-1)q^m}.$$

Proof. Suppose first that $B = B' + \langle b\rangle$ where b is a uniformly chosen random element of $\mathbb{F}_q^m$ and where $B' \supset T$ is a fixed space such that $\dim\langle AB'\rangle = \alpha(t+\beta-1)$. Let A^P be a projective version of A, meaning that for every $a \ne 0$ in A, we have exactly one element of the set

$$\{\lambda a, \lambda \in \mathbb{F}_q^*\}$$

in A^P.

We have $\dim\langle AB\rangle < \alpha(t+\beta-1)+\alpha$ if and only if the subspace bA has a non-zero intersection with $\langle AB'\rangle$, and also if and only if the set bA^P has a non-zero intersection with $\langle AB'\rangle$. Now,

$$\mathbb{P}\left(\dim\langle AB'\rangle \cap Ab \ne \{0\}\right) \le \sum_{a \in A^P, a \ne 0} \mathbb{P}\left(ab \in \langle AB'\rangle\right) \tag{8}$$

$$= \frac{|A|-1}{q-1}\frac{q^{\alpha(t+\beta-1)}}{q^m} \tag{9}$$

$$= \frac{q^{\alpha(t+\beta)}}{(q-1)q^m} - \frac{q^{\alpha(t+\beta-1)}}{(q-1)q^m}. \tag{10}$$

since for any fixed $a \ne 0$, we have that ab is uniformly distributed in $\mathbb{F}_q^m$, and since the number of elements in bA^P equals $(|A|-1)/(q-1)$.

Now write

$$B_0 = T \subset B_1 = T + \langle b_1\rangle \subset B_2 = T + \langle b_1, b_2\rangle \subset \cdots, \subset B_i = T + \langle b_1, \ldots, b_i\rangle \subset \cdots \subset B = B_\beta$$

where $b_1 \ldots, b_\beta$ are independent uniform vectors in $\mathbb{F}_q^m$. We have that the probability

$$\mathbb{P}\left(\dim\langle AB\rangle < \dim A \dim B\right)$$

that AB is not full-rank is not more than

$$\sum_{i=1}^{\beta} \mathbb{P}\left(\dim\langle AB_i\rangle < \dim A \dim B_i \mid \dim\langle AB_{i-1}\rangle = \dim A \dim B_{i-1}\right)$$

so that (10) gives:

$$\mathbb{P}\left(\dim\langle AB\rangle < \dim A \dim B\right) \le \frac{1}{q-1}\sum_{i=0}^{\beta-1}\left(\frac{1}{q^{m-(t+i+1)\alpha}} - \frac{1}{q^{m-(t+i)\alpha}}\right) \tag{11}$$

$$\le \frac{1}{q-1}\left(\frac{1}{q^{m-\alpha(t+\beta)}} - \frac{1}{q^{m-t\alpha}}\right) \le \frac{1}{(q-1)q^{m-\alpha(t+\beta)}}. \quad \square \tag{12}$$

B Proof of Theorem 2

Recall that a signature of a message M is a pair (x', y') where y' is a hashed value of the message M and $y' = H'x'^T$ and $\mathrm{rank}(x') = r$. If $\mathcal{A}$ is an algorithm that leads to a forgery with the use of N authentic signatures, then the algorithm $\mathcal{A}'$ consists of a simulated version of $\mathcal{A}$ where authentic signatures (x', y') are replaced by couples (x'', y'') where x'' is randomly and uniformly chosen among vectors of rank-weight e, and $y'' = H'x''^T$ is claimed to be the hashed value of the message M output by a random oracle. In the random oracle model, the algorithm $\mathcal{A}'$ must behave exactly as algorithm $\mathcal{A}$ and give the same output whenever (x'', y'') is statistically indistinguishable from (x', y').

We now compare the statistics of (x', y') and (x'', y''). We have $H' = A(R|H)P$ and since the transformation:

$$(x', y') \mapsto (x^a = Px', y^a = A^{-1}y') \tag{13}$$

$$(x'', y'') \mapsto (x^s = Px'', y^s = A^{-1}y'') \tag{14}$$

is one-to-one, comparing the statistics of (x', y') and (x'', y'') amounts to comparing the distributions of (x^a, y^a) (authentic) and (x^s, y^s) (simulated).

Now (x^a, y^a) is obtained in the following way: the signer chooses a subspace T of $\mathbb{F}_{q^m}$ together with a random vector $\tau \in T^t$ of rank t and is given a vector u which is uniformly distributed in the syndrome space $\mathbb{F}_{q^m}^{n-k}$ and is equal to $A^{-1}h(M) - R\tau^T$. Precisely, the signer chooses a random vector τ of rank t, and sets T to be the subspace generated by its coordinates. The signer then proceeds to try to decode u, meaning it looks for a subspace E of $\mathbb{F}_{q^m}$ that contains T and such that all coordinates of u fall into $\langle FE \rangle$, where F is the space generated by the elements of the LRPC matrix H. He succeeds exactly when the syndrome vector u is T-decodable in the sense of Definition 4: when this doesn't occur, the decoder aborts.

When the syndrome u is T-decodable, the decoder proceeds to solve the equation

$$Hx_H^T = u \tag{15}$$

and then sets

$$x^a = (\tau, x_H)$$

to create the couple $(x^a, y^a = (R|H)(x^a)^T)$ in (13). Recall from Remark 1 in Section 3.2 that the matrix H has been chosen so that equation (15) (equivalently equation (1)) always has a unique solution for every T-decodable u.

We may therefore speak about T-decodable *couples* (x_H, u), where u uniquely determines x_H and x_H uniquely determines u. Now, to re-cap, the authentic signer starts with uniformely random u, and whenever u turns out to be non T-decodable, then we declare a decoding failure and start the whole process again generating another τ, another random space T and another u by another call to the random oracle h (meaning a counter appended to the message M is incremented before applying the random hash again). This keeps happening until we hit a T-decodable u. We see therefore that when it does hit a T-decodable u, the couple

$$(x_H, u)$$

is uniformly distributed among all T-decodable couples.

We now turn to the action of the simulator: what the simulator does is he tries to generate a uniform T-decodable couple (x_H, u) through x_H rather than through u like the signer.

Specifically, the simulator starts with a random subspace E of $\mathbb{F}_{q^m}$ of dimension r and an x'' with coordinates independently and uniformly drawn from E. Since the transformation (14) $x'' \mapsto x^s = Px''$ is rank-preserving, the simulator is implicitely creating a uniform vector x^s of E^n. Write

$$x^s = (\tau, x_H).$$

With overwhelming probability (at least $1 - 1/q^{r-t}$), the vector $\tau \in E^t$ is of maximum rank-weight t, since its coordinates are independently and uniformly chosen in E. The vector τ generates the required random space T. Let $u = Hx_H^T$: note that by construction, all its coordinates must be in $\langle FE \rangle$. Consider the conditions (i),(ii),(iii) of Definition 4 for u to be T-decodable.

Remember that the first two conditions (i) and (ii) are properties *only of the subspace E*. When they are not satisfied, no choice of x_H can yield a T-decodable couple (x_H, u). When conditions (i) and (ii) are satisfied we have that, since the mapping (15) $x_H \mapsto u$ is invertible, the vector u is a uniform random vector in $\langle FE \rangle^{n-k}$. Since $n - k + \dim\langle FT \rangle = \dim\langle FE \rangle$, the probability that condition (iii) is not satisfied is governed by the probability than a random vector falls into a given subspace of $\langle FE \rangle$ of co-dimension 1 and is of the order of $1/q$: it is also at most $1/(q-1)$ according to computation (6). We also see that the number of x_H that satisfies condition (iii) is independent of the space E and is always the same for all E that satisfy conditions (i) and (ii). This last fact implies that when

- A random uniform subspace E is chosen among all possible subspaces E of dimension r,
- a random x_H is chosen in E^n,

then either (x_H, u) is not T-decodable, or (x_H, u) is T-decodable and is uniformly distributed among all T-decodable couples.

The simulator has no oracle to tell him when he has produced a non T-decodable couple, he can only hope this doesn't occur. As long as he produces T-decodable couples (x_H, u), then they are distributed (uniformly) exactly as those that are produced by the authentic signer and are undistinguishable from them. If we call π the probability that he produces a non-decodable u, then he can reasonably expect to produce a list of approximately $N = 1/\pi$ signatures (x'', y'') that are undistiguishable from genuine signatures in the random oracle model.

Consider now the probability π that the simulator produces a non-decodable u. It is at most the sum of the probabilities that E does not satisfy (i) and (ii) and the probability $\leq 1/(q-1)$ that (iii) is not satisfied. The probability that E does not satisfy (i) and (ii) is at most the probability that the product space $\langle (F_1^{-1}F + F_2^{-1})E \rangle$ does not have maximal dimension $(2d-1)r$, as argued in the proof of Theorem 1, and is at most $1/(q-1)$. We obtain therefore $\pi \leq 1/(q-1) + 1/(q-1) = 2/(q-1) \approx 2/q$ which concludes the proof.

C Other types of masking

Besides the augmented LRPC simple masking of Section 4, it is also possible to consider other types of masking, even if in that case there is no formal reduction, these masking have independent interest.

C.1 The $RankSign^\times$ scheme: multiplied LRPC codes

Let H be a parity-check matrix of an LRPC coder, for l a small fixed integer, we consider now M a random invertible $n \times n$ matrix of the form:

$$M = (M_1|M_2)$$

with M_1 a random $n \times l$ matrix over $GF(q^m)$ and M_2 a random $n \times (n-l)$ matrix in the base field $GF(q)$. Now consider the matrix

$$H'' = H.(M^{-1})^t$$

. We denote by *multiplied LRPC codes*, the codes with parity-check matrices of the form H''. The matrix H'' enables one to approximate a syndrome to a rank $r + t + l$ rather than $r + t$ with uniquely H. Hence the masking increases the rank of the small vector (remember that this rank has to be lower than the Singleton bound) but it considerably masks H. Indeed if M has a regular form, the inverse matrix M^{-1} is rather tricky and considerably masks H.

In that case one takes as public key: $H'' = AH.((MP)^{-1})^t$, suppose that for a given syndrome s we want to solve $H''.e'^t = s$, one proceeds as follows:

$$H''.e'^t = s \Leftrightarrow H.(e'(PM)^{-1})^T = A^{-1}s = s'$$

Rewriting the right handside as $H.e^t = s'$, we can decode s' in $e = e'(PM)^{-1}$ with $rank(e) = t + r$, and hence

$$e' = eMP$$

now since M has the particular form previously described, the rank of e' is $rank(e)+l = t+r+l$, the invertible matrix M does not change the weight.

Unlike the previous masking, this masking increases the obtained rank of the signature, and therefore makes the difference with the Singleton bound already big enough.

C.2 The $RankSign^{+\times}$ scheme: augmented-multiplied LRPC codes

This last masking combines the two previous masking scheme, we consider codes with parity-check matrices of the form:

$$H'.(M^{-1})^t$$

for H' an augmented matrix and M a matrix defined for multiplied LRPC, we denote these codes by *augmented-multiplied LRPC codes*. In the $RankSign^\times$ one starts from an augmented matrix H' as in $RankSign^+$ rather than an LRPC parity-check matrix H. The decoding algorithm mixes in a straightforward way the two previous approaches.

C.3   Parameters for *RankSign*$^\times$

- *RankSign*$^\times$

In that case the matrix H'' is a $(n-k) \times (n)$ matrix, and the rank of the signature is $r+t+l$.

n	n-k	m	q	t	r	r'	l	Singleton	public key	sign. size	security
40	20	39	2	3	10	13	2	19	15600	1185	100

- *RankSign*$^{+\times}$

In that case the matrix H''' is a $(n-k) \times (n+t')$ matrix (with t'=t), and the rank of the signature is $r+t+l$.

n	n-k	m	q	t	r	r'	l	Singleton	public key	sign. size	security
40	20	39	2	3	10	13	2	18	17940	1185	90

- *Cyclic* − *RankSign*$^{+\times}$

In that case the matrix H''' is a $(n-k) \times (n+t)$ quasi-cyclic matrix, the rank of the signature is $r+t+l$, and all implied matrices (A, P, R, M) are formed with concatenation of $\sigma \times \sigma$ cyclic blocks.

n	n-k	m	q	t	r	r'	l	σ	Singleton	public key	sign. size	security
40	20	36	2	2	10	12	2	2	17	7920	1092	90