# Recovering short secret keys of RLCE in polynomial time

Alain Couvreur*[1], Matthieu Lequesne†[2,3], and Jean-Pierre Tillich‡[2]

[1]Inria & LIX, CNRS UMR 7161
École polytechnique, 91128 Palaiseau Cedex, France.
[2]Inria, 2 rue Simone Iff, 75012 Paris, France.
[3]Sorbonne Université, UPMC Univ Paris 06

May 29, 2018

**Abstract**

We present a key recovery attack against Y. Wang's Random Linear Code Encryption (RLCE) scheme recently submitted to the NIST call for post-quantum cryptography. This attack recovers the secret key for all the short key parameters proposed by the author.

**Key words:** Code-based Cryptography, McEliece encryption scheme, key recovery attack, generalised Reed Solomon codes, Schur product of codes.

## Introduction

The McEliece encryption scheme dates back to the late 70's [14] and lies among the possible post-quantum alternatives to number theory based schemes using integer factorisation or discrete logarithm. However, the main drawback of McEliece's original scheme is the large size of its keys. Indeed, the classic instantiation of McEliece using binary Goppa codes requires public keys of several hundreds of kilobytes to assert a security of 128 bits. For example, the recent NIST submission *Classic McEliece* [4] proposes public keys of 1.1 to 1.3 megabytes to assert 256 bits security (with a classical computer).

For this reason, there is a recurrent temptation consisting in using codes with a higher decoding capacity for encryption in order to reduce the size of the public key. Many proposals in the last decades involve generalised Reed Solomon (GRS) codes, which are well–known to have a large minimum distance together with efficient decoding algorithms correcting up to half the minimum distance. On the other hand, the raw use of GRS codes has been proved to be insecure by Sidelnikov and Shestakov [15]. Subsequently, some variations have been proposed as a counter-measure of Sidelnikov and Shestakov's attack. Berger and Loidreau [3] suggested to replace a GRS code by a random subcode of small codimension, Wieschebrink

*alain.couvreur@lix.polytechnique.fr
†matthieu.lequesne@inria.fr
‡jean-pierre.tillich@inria.fr

[18] proposed to join random columns in a generator matrix of a GRS code and Baldi *et al.* [1] suggested to mask the structure of the code by right multiplying a generator matrix of a GRS code by the sum of a low rank matrix and a sparse matrix. It turns out that all of these proposals have been subject to efficient polynomial time attacks [19, 8, 11].

A more recent proposal by Yongge Wang [16] suggests another way of hiding the structure of GRS codes. The outline of Wang's construction is the following: start from a $k \times n$ generator matrix of a GRS code of length $n$ and dimension $k$ over a field $\mathbb{F}_q$, add $w$ additional random columns to the matrix, and mix the columns in a particular manner. The design of this scheme is detailed in § 2.1. This approach entails a significant expansion of the public key size but may resist above-mentioned attacks such as distinguisher and filtration attacks [8, 10]. This public key encryption primitive is the core of Wang's recent NIST submission "RLCE–KEM" [17].

**Our contribution** In the present article we give a polynomial time key recovery attack against RLCE which breaks the system when the number of additional random columns $w$ is strictly less than $n - k$. This allows us to break half the parameter sets proposed in [17].

# 1 Notation and prerequisites

## 1.1 Generalised Reed–Solomon codes

**Notation 1.** Let $q$ be a power of prime and $k$ a positive integer. We denote by $\mathbb{F}_q[X]_{<k}$ the vector space of polynomials over $\mathbb{F}_q$ whose degree is strictly bounded from above by $k$.

**Definition 2** (Generalised Reed Solomon codes)**.** Let $\boldsymbol{x} \in \mathbb{F}_q^n$ be a vector whose entries are pairwise distinct and $\boldsymbol{y} \in \mathbb{F}_q^n$ be a vector whose entries are all nonzero. The *generalised Reed–Solomon (GRS) code with support* $\boldsymbol{x}$ *and multiplier* $\boldsymbol{y}$ *of dimension* $k$ is defined as

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

## 1.2 Schur product of codes and square codes distinguisher

**Notation 3.** The component-wise product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ in $\mathbb{F}_q^n$ is denoted by

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

This definition extends to the product of codes where the *Schur product* of two codes $\mathscr{A}$ and $\mathscr{B} \subseteq \mathbb{F}_q^n$ is defined as

$$\mathscr{A} \star \mathscr{B} \stackrel{\text{def}}{=} \mathbf{Span}_{\mathbb{F}_q} \{\boldsymbol{a} \star \boldsymbol{b} \mid \boldsymbol{a} \in \mathscr{A},\ \boldsymbol{b} \in \mathscr{B}\}\,.$$

In particular, $\mathscr{A}^{\star 2}$ denotes the *square code* of a code $\mathscr{A}$: $\mathscr{A}^{\star 2} \stackrel{\text{def}}{=} \mathscr{A} \star \mathscr{A}$.

We recall the following result on the generic behaviour of random codes with respect to this operation.

**Proposition 4.** *([6, Theorem 2.3], informal) For a linear code* $\mathscr{R}$ *chosen at random over* $\mathbb{F}_q$ *of dimension* $k$ *and length* $n$*, the dimension of* $\mathscr{R}^{\star 2}$ *is typically* $\min(n, \binom{k+1}{2})$.

This provides a distinguisher between random codes and algebraically structured codes such as generalised Reed Solomon codes [19, 8], Reed Muller codes [7], polar codes [2] some Goppa codes [12, 10] or algebraic geometry codes [9]. For instance, in the case of GRS codes, we have the following result.

**Proposition 5.** *Let $n, k, \boldsymbol{x}, \boldsymbol{y}$ be as in Definition 2. Then,*

$$(\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}))^{\star 2} = \mathbf{GRS}_{2k-1}(\boldsymbol{x}, \boldsymbol{y} \star \boldsymbol{y}).$$

*In particular, if $k < n/2$, then*

$$\dim (\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}))^{\star 2} = 2k - 1.$$

Thus, compared to random codes whose square have dimension quadratic in the dimension of the code, the square of a GRS code has a dimension which is linear in that of the original code. This criterion allows to distinguish GRS codes of appropriate dimension from random codes.

### 1.3 Punctured and shortened codes

The notions of *puncturing* and *shortening* are classical ways to build new codes from existing ones. These constructions will be useful for the attack. We recall here their definition. Here, for a codeword $\boldsymbol{c} \in \mathbb{F}_q^n$, we denote $(c_1, \ldots, c_n)$ its entries.

**Definition 6** (punctured code)**.** Let $\mathscr{C} \subseteq \mathbb{F}_q^n$ and $\mathcal{L} \subseteq [\![1, n]\!]$. The *puncturing of $\mathscr{C}$ at $\mathcal{L}$* is defined as the code

$$\mathcal{P}_{\mathcal{L}}(\mathscr{C}) \stackrel{\text{def}}{=} \{(c_i)_{i \in [\![1,n]\!] \setminus \mathcal{L}} \text{ s.t. } \boldsymbol{c} \in \mathscr{C}\}.$$

A punctured code can be viewed as the restriction of the codewords to a subset of code positions. It will be sometimes more convenient to view a punctured code in this way. For this reason, we introduce the following definition.

**Definition 7** (restricted code)**.** Let $\mathscr{C} \subseteq \mathbb{F}_q^n$ and $\mathcal{L} \subseteq [\![1, n]\!]$. The restriction of $\mathscr{C}$ to $\mathcal{L}$ is defined as the code

$$\mathcal{R}_{\mathcal{L}}(\mathscr{C}) \stackrel{\text{def}}{=} \{(c_i)_{i \in \mathcal{L}} \text{ s.t. } \boldsymbol{c} \in \mathscr{C}\} = \mathcal{P}_{[\![1,n]\!] \setminus \mathcal{L}}(\mathscr{C}).$$

**Definition 8** (shortened code)**.** Let $\mathscr{C} \subseteq \mathbb{F}_q^n$ and $\mathcal{L} \subseteq [\![1, n]\!]$. The *shortening of $\mathscr{C}$ at $\mathcal{L}$* is defined as the code

$$\mathcal{S}_{\mathcal{L}}(\mathscr{C}) \stackrel{\text{def}}{=} \mathcal{P}_{\mathcal{L}}(\{\boldsymbol{c} \in \mathscr{C} \text{ s.t. } \forall i \in \mathcal{L},\ c_i = 0\}).$$

Shortening a code is equivalent to puncturing the dual code, as explained by the following proposition.

**Proposition 9** ([13, Theorem 1.5.7])**.** *Let $\mathscr{C}$ be a linear code over $\mathbb{F}_q^n$ and $\mathcal{L} \subseteq [\![1, n]\!]$. Then,*

$$\mathcal{S}_{\mathcal{L}}\left(\mathscr{C}^{\perp}\right) = (\mathcal{P}_{\mathcal{L}}(\mathscr{C}))^{\perp} \text{ and } (\mathcal{S}_{\mathcal{L}}(\mathscr{C}))^{\perp} = \mathcal{P}_{\mathcal{L}}\left(\mathscr{C}^{\perp}\right),$$

*where $\mathscr{A}^{\perp}$ denotes the dual of the code $\mathscr{A}$.*

**Notation 10.** Throughout the document, the indices of the columns (or positions of the codewords) will always refer to the indices in the original code, although the code has been punctured or shortened. For instance, consider a code $\mathscr{C}$ of length 5 where every word $\boldsymbol{c} \in \mathscr{C}$ is indexed $\boldsymbol{c} = (c_1, c_2, c_3, c_4, c_5)$. If we puncture $\mathscr{C}$ in $\{1, 3\}$, a codeword $\boldsymbol{c}' \in \mathcal{P}_{\{1,3\}}(\mathscr{C})$ will be indexed $(c'_2, c'_4, c'_5)$ and not $(c'_1, c'_2, c'_3)$.

## 2 The RLCE scheme

### 2.1 Presentation of the scheme

The RLCE encryption scheme is a code-based cryptosystem, inspired by the McEliece scheme. It has been introduced by Y. Wang in [16] and a proposal called "RLCE-KEM" has recently been submitted as a response for the NIST's call for post-quantum cryptosystems [17].

For a message $\boldsymbol{m} \in \mathbb{F}_q^k$, the cipher text is $\boldsymbol{c} = \boldsymbol{m}\boldsymbol{G} + \boldsymbol{e}$ where $\boldsymbol{e} \in \mathbb{F}_q^{n+w}$ is a random error vector of small weight $t$ and $\boldsymbol{G} \in \mathbb{F}_q^{k\times(n+w)}$ is a generator matrix defined as follows, for given parameters $n, k$ and $w$.

1. Let $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{F}_q^n$ be respectively a support and a multiplier (as in Definition 2).
2. Let $\boldsymbol{G}_0$ denote a $k \times n$ generator matrix of the generalised Reed–Solomon code $\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y})$ of length $n$ and dimension $k$. Denote by $g_1, \ldots, g_n$ the columns of $\boldsymbol{G}_0$.
3. Let $r_1, \ldots, r_w$ be column vectors chosen uniformly at random in $\mathbb{F}_q^k$. Denote by $\boldsymbol{G}_1$ the matrix obtained by inserting the random columns between GRS columns at the end of $\boldsymbol{G}_0$ as follows:
$$\boldsymbol{G}_1 \stackrel{\text{def}}{=} [g_1, \ldots, g_{n-w}, g_{n-w+1}, r_1, \ldots, g_n, r_w] \in \mathbb{F}_q^{k\times(n+w)}.$$
4. Let $\boldsymbol{A}_1, \ldots, \boldsymbol{A}_w$ be $2 \times 2$ matrices chosen uniformly at random in $\mathbf{GL}_2(\mathbb{F}_q)$. Let $\boldsymbol{A}$ be the block–diagonal non singular matrix
$$\boldsymbol{A} \stackrel{\text{def}}{=} \begin{pmatrix} \boldsymbol{I}_{n-w} & & & (0) \\ & \boldsymbol{A}_1 & & \\ & & \ddots & \\ (0) & & & \boldsymbol{A}_w \end{pmatrix} \in \mathbb{F}_q^{(n+w)\times(n+w)}.$$
5. Let $\pi \in \mathfrak{S}_{n+w}$ be a randomly chosen permutation of $[\![1, n+w]\!]$ and $\boldsymbol{P}$ the corresponding $(n+w) \times (n+w)$ permutation matrix.
6. The public key is the matrix $\boldsymbol{G} \stackrel{\text{def}}{=} \boldsymbol{G}_1\boldsymbol{A}\boldsymbol{P}$ and the private key is $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{A}, \boldsymbol{P})$.

**Note:** This is a slightly simplified version of the scheme proposed in [17] without the matrices $\boldsymbol{P}_1$ and $\boldsymbol{S}$ of the original description. They are actually not needed and the security of our simplified scheme is equivalent to the security of the scheme presented in [17].

### 2.2 Suggested sets of parameters

In [17] the author proposes 2 groups of 3 sets of parameters. The first group (referred to as *odd ID* parameters) corresponds to parameters such that $w \in [0.6(n-k), 0.7(n-k)]$, whereas in the second group (*even ID* parameters) the parameters satisfy $w = n - k$. The parameters of these two groups are listed in Tables 1 and 2.

The attack of the present paper recovers in polynomial time any secret key when parameters lie in the first group.

Table 1: Set of parameters for the first group : $w \in [0.6(n-k), 0.7(n-k)]$.

| Security level (bits) | Name in [17] | $n$ | $k$ | $t$ | $w$ | $q$ | Public key size (kB) |
|---|---|---|---|---|---|---|---|
| 128 | ID 1 | 532 | 376 | 78 | 96 | $2^{10}$ | 118 |
| 192 | ID 3 | 846 | 618 | 114 | 144 | $2^{10}$ | 287 |
| 256 | ID 5 | 1160 | 700 | 230 | 311 | $2^{11}$ | 742 |

Table 2: Set of parameters for the second group : $w = n - k$.

| Security level (bits) | Name in [17] | $n$ | $k$ | $t$ | $w$ | $q$ | Public key size (kB) |
|---|---|---|---|---|---|---|---|
| 128 | ID 0 | 630 | 470 | 80 | 160 | $2^{10}$ | 188 |
| 192 | ID 2 | 1000 | 764 | 118 | 236 | $2^{10}$ | 450 |
| 256 | ID 4 | 1360 | 800 | 280 | 560 | $2^{11}$ | 1232 |

# 3 Distinguishing by shortening and squaring

We will show here that it is possible to distinguish some public keys from random codes by computing the square of some shortening of the public code. More precisely, here is our main result.

**Theorem 11.** *Let $\mathscr{C}$ be a code over $\mathbb{F}_q$ of length $n+w$ and dimension $k$ with generator matrix $\boldsymbol{G}$ which is the public key of an RLCE scheme that is based on a GRS code of length $n$ and dimension $k$. Let $\mathcal{L} \subset [\![1, n+w]\!]$. Then,*

$$\dim\left(\mathcal{S}_{\mathcal{L}}\left(\mathscr{C}\right)\right)^{\star 2} \leqslant \min(n+w-|\mathcal{L}|,\ 2(k+w-|\mathcal{L}|)-1).$$

## 3.1 Restriction to the case where $\boldsymbol{P}$ is the identity

To prove Theorem 11 we can assume that $\boldsymbol{P}$ is the identity matrix. This is because of the following lemma.

**Lemma 12.** *For any permutation $\sigma$ of the code positions $[\![1, n+w]\!]$ we have*

$$\dim\left(\mathcal{S}_{\mathcal{L}}\left(\mathscr{C}\right)\right)^{\star 2} = \dim\left(\mathcal{S}_{\mathcal{L}^\sigma}\left(\mathscr{C}^\sigma\right)\right)^{\star 2},$$

*where $\mathscr{C}^\sigma$ is the set of codewords in $\mathscr{C}$ permuted by $\sigma$, that is $\mathscr{C}^\sigma = \{\boldsymbol{c}^\sigma : \boldsymbol{c} \in \mathscr{C}\}$ where $\boldsymbol{c}^\sigma \stackrel{def}{=} (c_{\sigma(i)})_{i \in [\![1,n+w]\!]}$ and $\mathcal{L}^\sigma \stackrel{def}{=} \{\sigma(i) : i \in \mathcal{L}\}$.*

Therefore, for the analysis of the distinguisher, we can make the following assumption.

**Assumption 13.** *The permutation matrix $\boldsymbol{P}$ is the identity matrix.*

We will use this assumption several times the rest of the section, especially to simplify the notation and define the terminology. The general case will follow by using Lemma 12.

## 3.2 Analysis of the different kinds of columns

### 3.2.1 Notation and terminology

Before proving the result, let us introduce some notation and terminology. The set of positions $[\![1, n+w]\!]$ splits in a natural way into four sets, whose definitions are given in the sequel

$$[\![1, n+w]\!] = \mathcal{I}^1_{\mathrm{GRS}} \cup \mathcal{I}^2_{\mathrm{GRS}} \cup \mathcal{I}_{\mathrm{R}} \cup \mathcal{I}_{\mathrm{PR}}. \tag{1}$$

**Definition 14.** The set of *GRS positions of the first type*, denoted $\mathcal{I}^1_{\mathrm{GRS}}$, corresponds to GRS columns which have not been associated to a random column. This set has cardinality $n-w$ and is given by

$$\mathcal{I}^1_{\mathrm{GRS}} \stackrel{\mathrm{def}}{=} \{i \in [\![1, n+w]\!] \,|\, \pi^{-1}(i) \leqslant n-w\}. \tag{2}$$

Under Assumption 13, this becomes:

$$\mathcal{I}^1_{\mathrm{GRS}} \stackrel{\mathrm{def}}{=} [\![1, n-w]\!].$$

This set is called this way, because at a position $i \in \mathcal{I}^1_{\mathrm{GRS}}$, any codeword $\boldsymbol{v} \in \mathscr{C}$ has an entry of the form

$$v_i = y_i f(x_i). \tag{3}$$

However, there might be other code positions that are of this form, as we will see later.

**Definition 15.** The set of *twin positions*, denoted $\mathcal{I}_{\mathrm{T}}$, corresponds to columns that result in a mix of a random column and a GRS one. This set has cardinality $2w$ and is equal to:

$$\mathcal{I}_{\mathrm{T}} \stackrel{\mathrm{def}}{=} \{i \in [\![1, n+w]\!] \,|\, \pi^{-1}(i) > n-w\}.$$

Under Assumption 13, this becomes:

$$\mathcal{I}_{\mathrm{T}} \stackrel{\mathrm{def}}{=} [\![n-w+1, n+w]\!].$$

The set $\mathcal{I}_{\mathrm{T}}$ can be divided in several subsets as follows.

**Definition 16.** Each position in $\mathcal{I}_{\mathrm{T}}$ has a unique corresponding *twin* position which is the position of the column with which it was mixed. For all $s \in [\![1, w]\!]$, $\pi(n-w+2s-1)$ and $\pi(n-w+2s)$ are twin positions. Under Assumption 13, the positions $n-w+2s-1$ and $n-w+2s$ are twins for all $s$ in $[\![1, w]\!]$.

For convenience, we introduce the following notation.

**Notation 17.** The twin of a position $i \in \mathcal{I}_{\mathrm{T}}$ is denoted by $\tau(i)$.

To any twin pair $\{i, \tau(i)\} = \{\pi(n-w+2s-1), \pi(n-w+2s)\}$ with $s \in \{1, \ldots, w\}$ is associated a unique linear form $\psi_s : \mathbb{F}_q[x]_{<k} \to \mathbb{F}_q$ and a non-singular matrix $\boldsymbol{A}_s$ such that for any codeword $\boldsymbol{v} \in \mathscr{C}$, we have

$$\begin{array}{rcl} v_i & = & a_s y_j f(x_j) + c_s \psi_s(f) \\ v_{\tau(i)} & = & b_s y_j f(x_j) + d_s \psi_s(f), \end{array} \tag{4}$$

where $j = n-w+s$ and

$$\begin{pmatrix} a_s & b_s \\ c_s & d_s \end{pmatrix} = \boldsymbol{A}_s. \tag{5}$$

The linear form $\psi_s$ is the form whose evaluations provides the random column added on the right of the $(n-w+s)$–th column during the construction process of $\boldsymbol{G}$ (see § 2.1, Step 3). From (4), we see that we may obtain more GRS positions: indeed $v_i = a_s y_j f(x_j)$ if $c_s = 0$ or $v_{\tau(i)} = b_s y_j f(x_j)$ if $d_s = 0$. On the other hand if $c_s d_s \neq 0$ the twin pairs are *correlated* in the sense that they behave in a non-trivial way after shortening: Lemma 24 shows that if one shortens the code in such a position its twin becomes a GRS position. We therefore call such a twin pair a *pseudo-random* twin pair and the set of pseudo-random twin pairs forms what we call the set of *pseudo-random* positions.

**Definition 18.** The set of *pseudo-random positions* (PR in short), denoted $\mathcal{I}_{\mathrm{PR}}$, is given by

$$\mathcal{I}_{\mathrm{PR}} \stackrel{\text{def}}{=} \bigcup_{s\in[\![1,w]\!] \ s.t. \ c_s d_s \neq 0} \{\pi(n-w+2s-1), \pi(n-w+2s)\}. \tag{6}$$

Under Assumption 13, this becomes:

$$\mathcal{I}_{\mathrm{PR}} = \bigcup_{s\in[\![1,w]\!] \ s.t. \ c_s d_s \neq 0} \{n-w+2s-1, n-w+2s\}. \tag{7}$$

If $c_s d_s = 0$, then a twin pair splits into a GRS position of the second kind and a random position. The GRS position of the second kind is $\pi(n-w+2s-1)$ if $c_s = 0$ or $\pi(n-w+2s)$ if $d_s = 0$ ($c_s$ and $d_s$ can not both be equal to 0 since $\boldsymbol{A}_s$ is invertible).

**Definition 19.** The set *GRS positions of the second kind*, denoted $\mathcal{I}^2_{\mathrm{GRS}}$, is defined as

$$\mathcal{I}^2_{\mathrm{GRS}} \stackrel{\text{def}}{=} \{\pi(n-w+2s-1) \,|\, c_s = 0\} \cup \{\pi(n-w+2s) \,|\, d_s = 0\}. \tag{8}$$

Under Assumption 13, this becomes:

$$\mathcal{I}^2_{\mathrm{GRS}} = \{n-w+2s-1 \,|\, c_s = 0\} \cup \{n-w+2s \,|\, d_s = 0\}. \tag{9}$$

**Definition 20.** The set of *random positions*, denoted $\mathcal{I}_{\mathrm{R}}$, is defined as

$$\mathcal{I}_{\mathrm{R}} \stackrel{\text{def}}{=} \{\pi(n-w+2s-1) \,|\, d_s = 0\} \cup \{\pi(n-w+2s) \,|\, c_s = 0\}. \tag{10}$$

Under Assumption 13, this becomes:

$$\mathcal{I}_{\mathrm{R}} = \{n-w+2s-1 \,|\, d_s = 0\} \cup \{n-w+2s \,|\, c_s = 0\}. \tag{11}$$

We also define the *GRS positions* to be the GRS positions of the first or the second kind.

**Definition 21.** The set of *GRS positions*, denoted $\mathcal{I}_{\mathrm{GRS}}$, is defined as

$$\mathcal{I}_{\mathrm{GRS}} \stackrel{\text{def}}{=} \mathcal{I}^1_{\mathrm{GRS}} \cup \mathcal{I}^2_{\mathrm{GRS}}. \tag{12}$$

*Remark* 22. Note that in the typical case $\mathcal{I}^2_{\mathrm{GRS}}$ and $\mathcal{I}_{\mathrm{R}}$ are empty sets. Indeed, such positions exist only if one of the entries (either $c_s$ or $d_s$) of a random non-singular matrix $\boldsymbol{A}_s$ is equal to zero.

We finish this subsection with a lemma.

**Lemma 23.** $|\mathcal{I}^2_{\mathrm{GRS}}| = |\mathcal{I}_{\mathrm{R}}|$ *and* $|\mathcal{I}_{\mathrm{PR}}| = 2(w - |\mathcal{I}_{\mathrm{R}}|)$.

*Proof.* Using (7), (9) and (11) we see that, under Assumption 13,

$$[\![n-w+1, n+w]\!] = \mathcal{I}_{\mathrm{PR}} \cup \mathcal{I}^2_{\mathrm{GRS}} \cup \mathcal{I}_{\mathrm{R}} \tag{13}$$

and the above union is disjoint. Next, there is a one-to-one correspondence relating $\mathcal{I}^2_{\mathrm{GRS}}$ and $\mathcal{I}_{\mathrm{R}}$. Indeed, still under Assumption 13, if $c_s = 0$ for some $s \in [\![1, w]\!]$, then $n-w+2s-1 \in \mathcal{I}^2_{\mathrm{GRS}}$ and $n-w+2s \in \mathcal{I}_{\mathrm{R}}$ and conversely if $d_s = 0$. This proves that $|\mathcal{I}^2_{\mathrm{GRS}}| = |\mathcal{I}_{\mathrm{R}}|$, which, together with (13) yields the result. □

### 3.3 Intermediate results

Before proceeding to the proof of Theorem 11, let us state and prove some intermediate results. We will start by Lemmas 24 and 26, that will be useful to prove Proposition 27 on the structure of shortened RLCE codes, by induction on the number of shortened positions. This proposition will be the key of the final theorem. Then, we will prove a general result on modified GRS codes with additional random columns.

#### 3.3.1 Two useful lemmas

The first lemma explains that, after shortening a PR position, its twin will behave like a GRS position. This is actually a crucial lemma that explains why PR columns in $\boldsymbol{G}$ do not really behave like random columns after shortening the code at the corresponding position.

**Lemma 24.** *Let $i$ be a PR position and $\mathcal{L}$ a set of positions that neither contains $i$ nor $\tau(i)$. Let $\mathscr{C}' \stackrel{def}{=} \mathcal{S}_{\mathcal{L}}(\mathscr{C})$. The position $\tau(i)$* behaves like a GRS position *in the code $\mathcal{S}_{\{i\}}(\mathscr{C}')$. That is, the $\tau(i)$–th column of a generator matrix of $\mathcal{S}_{\{i\}}(\mathscr{C}')$ has entries of the form*

$$\tilde{y}_j f(x_j)$$

*for some $j$ in $[\![n-w+1, n]\!]$ and $\tilde{y}_j$ in $\mathbb{F}_q$.*

*Proof.* Let us assume that $i = n-w+2s-1$ for some $s \in \{1, \ldots, w\}$. The case $i = n-w+2s$ can be proved in a similar way. At position $i$, for any $\boldsymbol{c} \in \mathscr{C}'$, from (4), we have

$$c_i = a y_j f(x_j) + c\psi_s(f),$$

where $j = n - w + s$. By shortening, we restrict our space of polynomials to the subspace of polynomials in $\mathbb{F}_q[x]_{<k}$ satisfying $c_i = 0$, *i.e.* Since $i$ is a PR position, $c \neq 0$ and therefore

$$\psi_s(f) = -c^{-1} a y_j f(x_j).$$

Therefore, at the twin position $\tau(i) = n - w + 2s$ and for any $\boldsymbol{c} \in \mathcal{S}_{\{i\}}(\mathscr{C}')$, we have

$$\begin{aligned} c_{\tau(i)} &= b y_j f(x_j) + d\psi_j(f) \\ &= y_j (b - dac^{-1}) f(x_j). \end{aligned}$$

□

*Remark* 25. This lemma does not hold for a random position, since the proof requires that $c \neq 0$. It is precisely because of this that we have to make a distinction between twin pairs, *i.e.* pairs for which the associated matrix $\boldsymbol{A}_s$ is such that $c_s d_s \neq 0$ and pairs for which it is not the case.

This lemma allows us to get some insight on the structure of the shortened code $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$. Before giving the relevant statement let us first recall the following result.

**Lemma 26.** *Consider a linear code $\mathscr{A}$ over $\mathbb{F}_q$ whose restriction to a subset $\mathcal{L}$ is a subcode of a GRS code over $\mathbb{F}_q$ of dimension $k_{\mathrm{GRS}}$. Let $i$ be an element of $\mathcal{L}$. Then the restriction of $\mathcal{S}_{\{i\}}(\mathscr{A})$ to $\mathcal{L} \setminus \{i\}$ is a subcode of a GRS code of dimension $k_{\mathrm{GRS}} - 1$.*

*Proof.* By definition the restriction $\mathscr{A}'$ to $\mathcal{L}$ is a code of the form

$$\mathscr{A}' \stackrel{\text{def}}{=} \left\{ \left(y_j f(x_j)\right)_{j\in\mathcal{L}} : f \in L \right\},$$

where the $y_j$'s are nonzero elements of $\mathbb{F}_q$, the $x_j$'s are distinct elements of $\mathbb{F}_q$ and $L$ is a subspace of $\mathbb{F}_q[X]_{<k_{\text{GRS}}}$. Clearly the restriction $\mathscr{A}''$ of $\mathcal{S}_{\{i\}}(\mathscr{A})$ to $\mathcal{L}\setminus\{i\}$ can be written as

$$\mathscr{A}'' = \left\{ \left(y_j f(x_j)\right)_{j\in\mathcal{L}\setminus\{i\}} : f \in L, f(x_i) = 0 \right\}.$$

The polynomials $f(X)$ in $L$ such that $f(x_i) = 0$ can be written as $f(X) = (X - x_i)g(X)$ where $\deg g = \deg f - 1$ and $g$ ranges in this case over a subspace $L'$ of polynomials of degree $< k_{\text{GRS}} - 1$. We can therefore write

$$\mathscr{A}'' = \left\{ \left(y_j(x_j - x_i)g(x_j)\right)_{j\in\mathcal{L}\setminus\{i\}} : g \in L' \right\}.$$

This implies our lemma. □

### 3.3.2 The key proposition

Using Lemmas 24 and 26, we can prove the following result by induction. This result is the key proposition for proving Theorem 11.

**Proposition 27.** *Let $\mathcal{L}$ be a subset of $[\![1, n+w]\!]$ and let $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2$ be subsets of $\mathcal{L}$ defined as*

- *$\mathcal{L}_0$ the set of GRS positions (see (2), (8) and (12) for a definition) of $\mathcal{L}$, i.e.*

$$\mathcal{L}_0 \stackrel{def}{=} \mathcal{L} \cap \mathcal{I}_{\text{GRS}};$$

- *$\mathcal{L}_1$ the set of PR positions (see (6)) of $\mathcal{L}$ that do not have their twin in $\mathcal{L}$, i.e.*

$$\mathcal{L}_1 \stackrel{def}{=} \{i \in \mathcal{L} \cap \mathcal{I}_{\text{PR}} \,|\, \tau(i) \notin \mathcal{L}\};$$

- *$\mathcal{L}_2$ the set of PR positions of $\mathcal{L}$ whose twin position is also included in $\mathcal{L}$, i.e.*

$$\mathcal{L}_2 \stackrel{def}{=} \{i \in \mathcal{L} \cap \mathcal{I}_{\text{PR}} \,|\, \tau(i) \in \mathcal{L}\}.$$

*Let $\mathscr{C}'$ be the restriction of $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$ to $(\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}_0) \cup \tau(\mathcal{L}_1)$. Then, $\mathscr{C}'$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$.*

*Proof.* Let us prove by induction on $\ell = |\mathcal{L}|$ that $\mathscr{C}'$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$.

This statement is clearly true if $\ell = 0$, *i.e.* if $\mathcal{L}$ is the empty set. Assume that the result is true for all $\mathcal{L}$ up to some size $\ell \geqslant 0$. Consider now a set $\mathcal{L}$ of size $\ell + 1$. We can write $\mathcal{L} = \mathcal{L}' \cup \{i\}$ where $\mathcal{L}'$ is of size $\ell$.

Let $\mathcal{L}_0, \mathcal{L}_1, \mathcal{L}_2$ be subsets of $\mathcal{L}$ as defined in the statement and $\mathcal{L}'_0, \mathcal{L}'_1, \mathcal{L}'_2$ be the subsets of $\mathcal{L}'$ obtained by replacing in the statement $\mathcal{L}$ by $\mathcal{L}'$. There are now several cases to consider for $i$.

**Case 1:** $i \in \mathcal{L}_0$. In this case, $\mathcal{L}_0 = \mathcal{L}'_0 \cup \{i\}$, $\mathcal{L}_1 = \mathcal{L}'_1$ and $\mathcal{L}_2 = \mathcal{L}'_2$. We can apply Lemma 26 with $\mathscr{A} = \mathcal{S}_{\mathcal{L}'}(\mathscr{C})$ because by the induction hypothesis, its restriction to

$$\mathcal{L}'' \stackrel{\text{def}}{=} (\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}'_0) \cup \tau(\mathcal{L}'_1)$$

is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}'_0| + |\mathcal{L}'_1|$ and dimension $k - |\mathcal{L}'_0| - \frac{|\mathcal{L}'_2|}{2}$. Therefore the restriction of the shortened code $\mathcal{S}_{\mathcal{L}}(\mathscr{C}) = \mathcal{S}_{\{i\}}(\mathscr{A})$ to $\mathcal{L}'' \setminus \{i\} = (\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}_0) \cup \tau(\mathcal{L}_1)$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k - |\mathcal{L}'_0| - \frac{|\mathcal{L}'_2|}{2} - 1 = k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$.

**Case 2:** $i \in \mathcal{L}_1$. In this case, $\mathcal{L}_0 = \mathcal{L}'_0, \mathcal{L}_1 = \mathcal{L}'_1 \cup \{i\}$ and $\mathcal{L}_2 = \mathcal{L}'_2$. This implies that $\mathcal{L}'$ does not contain $i$ nor $\tau(i)$. We can therefore apply Lemma 24 with $\mathscr{C}' = \mathcal{S}_{\mathcal{L}'}(\mathscr{C})$. Lemma 24 states that the position $\tau(i)$ behaves like a GRS position in $\mathcal{S}_{\{i\}}(\mathscr{C}') = \mathcal{S}_{\mathcal{L}}(\mathscr{C})$. By induction hypothesis, the restriction of the code $\mathscr{C}'$ to $(\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}'_0) \cup \tau(\mathcal{L}'_1)$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}'_0| + |\mathcal{L}'_1|$ and dimension $k - |\mathcal{L}'_0| - \frac{|\mathcal{L}'_2|}{2} = k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$. Therefore the restriction of $\mathcal{S}_{\{i\}}(\mathscr{C}') = \mathcal{S}_{\mathcal{L}}(\mathscr{C})$ to $(\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}_0) \cup \tau(\mathcal{L}_1) = (\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}'_0) \cup \tau(\mathcal{L}'_1) \cup \{\tau(i)\}$ is a subcode of a GRS code of dimension $k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$ and length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}'_0| + |\mathcal{L}'_1| + 1 = |\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$.

**Case 3:** $i \in \mathcal{L}_2$. In this case, $\mathcal{L}_0 = \mathcal{L}'_0, \mathcal{L}_1 = \mathcal{L}'_1 \setminus \{\tau(i)\}$ and $\mathcal{L}_2 = \mathcal{L}'_2 \cup \{i, \tau(i)\}$. In fact, this case can only happen if $\ell \geqslant 1$ and we will rather consider the induction with respect to the set $\mathcal{L}'' = \mathcal{L} \setminus \{i, \tau(i)\}$ of size $\ell - 1$ and the sets $\mathcal{L}''_0, \mathcal{L}''_1, \mathcal{L}''_2$ such that $\mathcal{L}''_0 = \mathcal{L}_0, \mathcal{L}''_1 = \mathcal{L}_1, \mathcal{L}''_2 = \mathcal{L}_2 \setminus \{i, \tau(i)\}$.

By induction hypothesis on $\mathcal{L}''$, the restriction of $\mathscr{C}'' \stackrel{\text{def}}{=} \mathcal{S}_{\mathcal{L}''}(\mathscr{C})$ to $(\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}''_0) \cup \tau(\mathcal{L}''_1)$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}''_0| + |\mathcal{L}''_1| = |\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k - |\mathcal{L}''_0| - \frac{|\mathcal{L}''_2|}{2} = k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2} + 1$.

Following Assumption 13, we can write without loss of generality that $i = n - w + 2s - 1$ for some $s \in \{1, \ldots, w\}$. The case $i = n - w + 2s$ can be proved in a similar way. Denote $\boldsymbol{A}_s = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ the non-singular matrix and $j = n - w + s$.

For any $\boldsymbol{c} \in \mathscr{C}'$, at positions $i$ and $\tau(i)$ we have

$$\begin{aligned} c_i &= a y_j f(x_j) + c\psi_s(f), \\ c_{\tau(i)} &= b y_j f(x_j) + d\psi_s(f). \end{aligned}$$

Shortening $\mathscr{C}''$ at $\{i, \tau(i)\}$ has the effect of requiring to consider only the polynomials $f$ for which $f(x_j) = \psi_s(f) = 0$. Therefore the restriction of $\mathcal{S}_{\{i,\tau(i)\}}(\mathscr{C}'') = \mathcal{S}_{\mathcal{L}}(\mathscr{C})$ at $(\mathcal{I}_{\text{GRS}} \setminus \mathcal{L}''_0) \cup \tau(\mathcal{L}''_1)$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2} + 1 - 1 = k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$.

**Case 4:** $i \in \mathcal{I}_{\text{R}}$. In this case $\mathcal{L}_0 = \mathcal{L}'_0, \mathcal{L}_1 = \mathcal{L}'_1$ and $\mathcal{L}_2 = \mathcal{L}'_2$. Using the induction hypothesis yields directly that $\mathscr{A} = \mathcal{S}_{\mathcal{L}'}(\mathscr{C})$ is a subcode of a GRS code of length $|\mathcal{I}_{\text{GRS}}| - |\mathcal{L}'_0| + |\mathcal{L}'_1| = |\mathcal{I}_{\text{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k - |\mathcal{L}'_0| - \frac{|\mathcal{L}'_2|}{2} = k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$. This is also clearly the case for $\mathcal{S}_{\mathcal{L}}(\mathscr{C}) = \mathcal{S}_{\{i\}}(\mathscr{A})$.

This proves that the induction hypothesis also holds for $|\mathcal{L}| = \ell + 1$ and finishes the proof of the proposition. □

### 3.3.3 A general result on modified GRS codes

Finally, we need a very general result concerning modified GRS codes where some arbitrary columns have been joined to the generator matrix. A very similar lemma is already proved in [8, Lemma 9]. Its proof is repeated below for convenience and in order to provide further details about the equality case.

**Lemma 28.** *Consider a linear code $\mathscr{A}$ over $\mathbb{F}_q$ with generator matrix $\boldsymbol{G} = \begin{pmatrix} \boldsymbol{G}_{\mathrm{SCGRS}} & \boldsymbol{G}_{\mathrm{rand}} \end{pmatrix} \boldsymbol{P}$ of size $k \times (n+r)$ where $\boldsymbol{G}_{\mathrm{SCGRS}}$ is a $k \times n$ generator matrix of a subcode of a GRS code of dimension $k_{\mathrm{GRS}}$ over $\mathbb{F}_q$, $\boldsymbol{G}_{\mathrm{rand}}$ is an arbitrary matrix in $\mathbb{F}_q^{k\times r}$ and $\boldsymbol{P}$ is the permutation matrix of an arbitrary permutation $\sigma \in \mathfrak{S}_{n+r}$. We have*

$$\dim \mathscr{A}^{\star 2} \leqslant 2k_{\mathrm{GRS}} - 1 + r.$$

*Moreover, if the equality holds, then for every $i \in [\![n+1, n+w]\!]$ we have:*

$$\dim \mathcal{P}_{\{\sigma(i)\}}\left(\mathscr{A}^{\star 2}\right) = \dim \mathscr{A}^{\star 2} - 1.$$

*Proof.* Without loss of generality, we may assume that $\boldsymbol{P}$ is the identity matrix since the dimension of the square code is invariant by permuting the code positions, as seen in Lemma 12. Let $\mathscr{B}$ be the code with generator matrix $\begin{pmatrix} \boldsymbol{G}_{\mathrm{SCGRS}} & \boldsymbol{0}_{k\times r} \end{pmatrix}$ where $\boldsymbol{0}_{k\times r}$ is the zero matrix of size $k \times r$. We also define the code $\mathscr{B}'$ generated by the generator matrix $\begin{pmatrix} \boldsymbol{0}_{k\times n} & \boldsymbol{G}_{\mathrm{rand}} \end{pmatrix}$. We obviously have

$$\mathscr{A} \subseteq \mathscr{B} + \mathscr{B}'.$$

Therefore

$$\begin{aligned} (\mathscr{A})^{\star 2} &\subseteq \left(\mathscr{B} + \mathscr{B}'\right)^{\star 2} \\ &\subseteq \mathscr{B}^{\star 2} + \left(\mathscr{B}'\right)^{\star 2} + \mathscr{B} \star \mathscr{B}' \\ &\subseteq \mathscr{B}^{\star 2} + \left(\mathscr{B}'\right)^{\star 2}, \end{aligned}$$

where the last inclusion comes from the fact that $\mathscr{B} \star \mathscr{B}'$ is the zero subspace since $\mathscr{B}$ and $\mathscr{B}'$ have disjoint supports. The code $\mathscr{B}^{\star 2}$ has dimension $\leqslant 2k_{\mathrm{GRS}} - 1$ whereas $\dim \left(\mathscr{B}'\right)^{\star 2} \leqslant r$.

Next, if $\dim \mathscr{A}^{\star 2} = 2k_{\mathrm{GRS}} - 1 + r$, then

$$\mathscr{A}^{\star 2} = \mathscr{B}^{\star 2} \oplus (\mathscr{B}')^{\star 2} \quad \text{and} \quad \dim(\mathscr{B}')^{\star 2} = r.$$

Since $\mathscr{B}'$ has length $r$, this means that $(\mathscr{B}')^{\star 2} = \mathbb{F}_q^r$ and hence, any word of weight 1 supported by the $r$ rightmost positions is contained in $\mathscr{A}^{\star 2}$. Therefore, puncturing this position will decrease the dimension. □

## 3.4 Proof of the theorem

We are now ready to prove Theorem 11.

*Proof of Theorem 11.* By using Proposition 27, we know that the restriction of $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$ to $(\mathcal{I}_{\mathrm{GRS}} \setminus \mathcal{L}_0) \cup \tau(\mathcal{L}_1)$ is a subcode of a GRS code of length $|\mathcal{I}_{\mathrm{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1| = n - w + |\mathcal{I}^2_{\mathrm{GRS}}| - |\mathcal{L}_0| + |\mathcal{L}_1|$ and dimension $k_{\mathrm{GRS}} \stackrel{\mathrm{def}}{=} k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}$, where:

- $\mathcal{L}_0 \stackrel{\mathrm{def}}{=} \mathcal{I}_{\mathrm{GRS}} \cap \mathcal{L}$;

- $\mathcal{L}_1$ is the set of PR positions of $\mathcal{L}$ that do not have their twin in $\mathcal{L}$;
- $\mathcal{L}_2$ is the union of all twin PR positions that are both included in $\mathcal{L}$.

We also denote by $\mathcal{L}_3$ the set $\mathcal{I}_{\mathrm{R}} \cap \mathcal{L}$. We can then apply Lemma 28 to $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$ and derive from it the following upper bound:

$$\dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} \leqslant 2k_{\mathrm{GRS}} - 1 + |\mathcal{I}_{\mathrm{PR}} \setminus (\mathcal{L} \cup \tau(\mathcal{L}_1))| + |\mathcal{I}_{\mathrm{R}} \setminus \mathcal{L}_3|.$$

Next, using Lemma 23, we get

$$\begin{aligned}
\dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} &\leqslant 2\left(k - |\mathcal{L}_0| - \frac{|\mathcal{L}_2|}{2}\right) - 1 + 2\left(w - |\mathcal{I}_{\mathrm{R}}|\right) - 2|\mathcal{L}_1| - |\mathcal{L}_2| + |\mathcal{I}_{\mathrm{R}}| - |\mathcal{L}_3| \\
&\leqslant 2\left(k + w - |\mathcal{L}_0| - |\mathcal{L}_1| - |\mathcal{L}_2| - |\mathcal{L}_3|\right) - 1 + \left(|\mathcal{L}_3| - |\mathcal{I}_{\mathrm{R}}|\right) \quad (14) \\
&\leqslant 2\left(k + w - |\mathcal{L}|\right) - 1. \quad (15)
\end{aligned}$$

The other upper bound on $\dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2}$ which is $\dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} \leqslant n + w - |\mathcal{L}|$ follows from the fact that the dimension of this code is bounded by its length. Putting both bounds together yields the theorem. □

*Remark* 29. According to our simulations, the inequality of Theorem 11 is sharp and is attained most of the time when $\mathcal{I}_{\mathrm{R}}$ is the empty set. When $\mathcal{I}_{\mathrm{R}}$ is not the empty set, then we may have equality only if we shorten all the positions in $\mathcal{I}_{\mathrm{R}}$: this is because the right-hand term in (14) should coincide with the right-hand term in (15), which is equivalent to $|\mathcal{L}_3| = |\mathcal{I}_{\mathrm{R}}|$.

# 4 Reaching the range of the distinguisher

For this distinguisher to work we need to shorten the code enough so that its square does not fill in the ambient space, but not too much since the square of the shortened code should have a dimension strictly less than the typical dimension of the square of a random code given by Proposition 4. Namely, we need to have:

$$\dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} < \binom{k+1-|\mathcal{L}|}{2} \qquad \text{and} \qquad \dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} < n + w - |\mathcal{L}|. \quad (16)$$

Thanks to Theorem 11, we know that (16) is satisfied as soon as

$$2(k + w - |\mathcal{L}|) - 1 < \binom{k+1-|\mathcal{L}|}{2} \qquad \text{and} \qquad 2(k + w - |\mathcal{L}|) - 1 < n + w - |\mathcal{L}|. \quad (17)$$

We will now find the values of $|\mathcal{L}|$ for which both inequalities of (17) are satisfied.

**First inequality.** To study when the first inequality in (17) is verified, let us bring in

$$k' \stackrel{\text{def}}{=} k - |\mathcal{L}|.$$

Inequality (17) becomes $4k' - 2 + 4w < k'^2 + k'$, or equivalently $k'^2 - 3k' - 4w + 2 > 0$, which after a resolution leads to $k' > \frac{3+\sqrt{16w+1}}{2}$.

Hence, we have:

$$|\mathcal{L}| < k - \frac{3 + \sqrt{16w+1}}{2}. \quad (18)$$

**Second inequality.** On the other hand, the second inequality in (17) is equivalent to

$$|\mathcal{L}| \geqslant w + 2k - n. \tag{19}$$

**Conditions to verify both inequalities.** Putting inequalities (18) and (19) together gives that $|\mathcal{L}|$ should satisfy

$$w + 2k - n \leqslant |\mathcal{L}| < k - \frac{3 + \sqrt{16w+1}}{2}.$$

We can therefore find an appropriate $\mathcal{L}$ if and only if

$$w + 2k - n < k - \frac{3 + \sqrt{16w+1}}{2},$$

which is equivalent to

$$n - k > w + \frac{3 + \sqrt{16w+1}}{2} = w + O(\sqrt{w}).$$

In other words, the distinguisher works up to values of $w$ that are close to the second choice $n - k = w$.

From now on we set

$$\begin{aligned} \ell_{\min} &= w + 2k - n \\ \ell_{\max} &= \left\lceil k - \frac{3 + \sqrt{16w+1}}{2} - 1 \right\rceil. \end{aligned}$$

**Practical results.** We have run experiments using Magma [5] and Sage. For the parameters of Table 1, here are the intervals of possible values of $|\mathcal{L}|$ so that the code $\mathcal{S}_{\mathcal{L}}(\mathscr{C})^{\star 2}$ has a non generic dimension:

- ID 1: $n = 532, k = 376, w = 96$, $|\mathcal{L}| \in [\![316, 354]\!]$;
- ID 3: $n = 846, k = 618, w = 144$, $|\mathcal{L}| \in [\![534, 592]\!]$;
- ID 5: $n = 1160, k = 700, w = 311$, $|\mathcal{L}| \in [\![551, 663]\!]$.

There interval always coincide with the theoretical interval $[\![\ell_{\min}, \ell_{\max}]\!]$.

# 5 The attack

In this section we will show how to find an equivalent private key $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{A}, \boldsymbol{P})$ defining the same code. This allows to decode and recover the original message like a legitimate user.

We assume that all the matrices $\boldsymbol{A}_s = \begin{pmatrix} a_s & b_s \\ c_s & d_s \end{pmatrix}$ appearing in the definition of the scheme in Subsection 2.1 are such that $c_s d_s \neq 0$. We explain in the appendix how the attack can be changed to take the case $c_s d_s = 0$ into account. Note that this corresponds to a case where $\mathcal{I}_{\mathrm{R}} = \emptyset$ and $\mathcal{I}^2_{\mathrm{GRS}} = \emptyset$, which is the typical case as noticed in Remark 22.

*Remark* 30. In the present section where we the goal is to recover the permutation, we no longer work under Assumption 13.

### 5.1 Outline of the attack

In summary, the attack works as follows.

1. Compute the interval $[\![\ell_{\min}, \ell_{\max}]\!]$ of the distinguisher and choose $\ell$ in the middle of the distinguisher interval. Ensure $\ell < \ell_{\max}$.
2. For several sets of indices $\mathcal{L} \subseteq [\![1, n+w]\!]$ such that $|\mathcal{L}| = \ell$, compute $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$ and identify pairs of twin positions contained in $[\![1, n+w]\!]$. Repeat this process until identifying all pairs of twin positions, as detailed in § 5.2.
3. Puncture the twin positions in order to get a GRS code and recover its structure using the Sidelnikov Shestakov attack [15].
4. For each pair of twin positions, recover the corresponding $2 \times 2$ non-singular matrix $A_i$, as explained in § 5.4.
5. Finish to recover the structure of the underlying GRS code.

### 5.2 Identifying pairs of twin positions

Let $\mathcal{L} \subseteq [\![1, n+w]\!]$ be such that both $|\mathcal{L}|$ and $|\mathcal{L}|+1$ are contained in the distinguisher interval. The idea we use to identify pairs of twin positions is to compare the dimension of $(\mathcal{S}_{\mathcal{L}}(\mathscr{C}))^{\star 2}$ with the dimension of $\left(\mathcal{P}_{\{i\}}(\mathcal{S}_{\mathcal{L}}(\mathscr{C}))\right)^{\star 2}$ for all positions $i$ in $[\![1, n+w]\!] \setminus \mathcal{L}$. This yields information on pairs of twin positions.

- If $i \in \mathcal{I}_{\mathrm{GRS}}$ (see (2), (8) and (12) for the definition), puncturing does not affect the dimension of the square code:

$$\dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} = \dim\left(\mathcal{P}_{\{i\}}\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)\right)^{\star 2}.$$

- If $i \in \mathcal{I}_{\mathrm{PR}}$ (see (6) for a definition) and $\tau(i) \in \mathcal{L}$, then according to Lemma 24, the position $i$ is "derandomised" in $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$ and hence behaves like a GRS position in the shortened code. Therefore, very similarly to the previous case, the dimension does not change.

- If $i \in \mathcal{I}_{\mathrm{PR}}$ and $\tau(i) \notin \mathcal{L}$, in $\mathcal{S}_{\mathcal{L}}(\mathscr{C})$, the two corresponding columns behave like random ones. Assuming that the inequality of Theorem 11 is an equality, which almost always holds (see Remark 29), then, according to Lemma 28, puncturing $\mathcal{S}_{\mathcal{L}}(\mathscr{C})^{\star 2}$ at $i$ (resp. $\tau(i)$) reduces its dimension. Therefore,

$$\dim\left(\mathcal{P}_{\{i\}}\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)\right)^{\star 2} = \dim\left(\mathcal{P}_{\{\tau(i)\}}\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)\right)^{\star 2} = \dim\left(\mathcal{S}_{\mathcal{L}}(\mathscr{C})\right)^{\star 2} - 1.$$

This provides a way to identify any position in $[\![1, n+w]\!] \setminus \mathcal{L}$ having a twin which also lies in $[\![1, n+w]\!] \setminus \mathcal{L}$: by searching zero columns in a parity–check matrix of $\mathcal{S}(\mathscr{C})^{\star 2}$, we obtain the set $\mathcal{T}_{\mathcal{L}} \subset [\![1, n+w]\!] \setminus \mathcal{L}$ of even cardinality of all the positions having their twin in $[\![1, n+w]\!] \setminus \mathcal{L}$:

$$\mathcal{T}_{\mathcal{L}} \stackrel{\text{def}}{=} \bigcup_{\{i,\tau(i)\} \subseteq [\![1,n+w]\!] \setminus \mathcal{L}} \{i, \tau(i)\}.$$

As soon as these positions are identified, we can associate each such position to its twin. This can be done as follows. Take $i \in \mathcal{T}_{\mathcal{L}}$ and consider the code $\mathcal{S}_{\mathcal{L}\cup\{i\}}(\mathscr{C})$. The column corresponding to the twin position $\tau(i)$ has been derandomised and hence will not give a zero column in a parity–check matrix of $\left(\mathcal{S}_{\mathcal{L}\cup\{i\}}(\mathscr{C})\right)^{\star 2}$, so puncturing the corresponding column will not affect the dimension.

This process can be iterated by using various shortening sets $\mathcal{L}$ until obtaining $w$ pairs of twin positions. It is readily seen that considering $O(1)$ such sets is enough to recover all pairs with very large probability.

## 5.3 Recovering the non-randomised part of the code

As soon as all the pairs of twin positions are identified, consider the code $\mathcal{P}_{\mathcal{I}_{\rm PR}}(\mathscr{C})$ punctured at $\mathcal{I}_{\rm PR}$. Since the randomised positions have been punctured this code is nothing but a GRS code and, applying the Sidelnikov Shestakov attack [15], we recover a pair $\boldsymbol{a}, \boldsymbol{b}$ such that

$$\mathcal{P}_{\mathcal{I}_{\rm PR}}(\mathscr{C}) = \mathbf{GRS}_k(\boldsymbol{a}, \boldsymbol{b}).$$

## 5.4 Recovering the remainder of the code and the matrix $\boldsymbol{A}$

### 5.4.1 Joining a pair of twin positions : the code $\mathscr{C}^{(i)}$

To recover the remaining part of the code we will consider iteratively the pairs of twin positions. We recall that $\mathcal{I}_{\rm PR}$ corresponds to the set of positions having a twin. Let $\{i, \tau(i)\}$ be a pair of twin positions and consider the code

$$\mathscr{C}^{(i)} \stackrel{\rm def}{=} \mathcal{R}_{\mathcal{I}_{\rm GRS}\cup\{i,\tau(i)\}}(\mathscr{C}).$$

In this code, all but two columns, columns $i$ and $\tau(i)$, are GRS positions.

For any codeword $\boldsymbol{c} \in \mathscr{C}^{(i)}$ we have

$$\begin{array}{rcl} c_i & = & a y_j f(x_j) + c\psi_j(f) \\ c_{\tau(i)} & = & b y_j f(x_j) + d\psi_j(f) \end{array} \tag{20}$$

for some $j \in [\![n-w+1, n]\!]$, where $\psi_j$ and $\boldsymbol{A} = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ are defined as in (4) and (5).

Note that we do not need to recover exactly $(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{A}, \boldsymbol{P})$. We need to recover a 4–tuple $(\boldsymbol{x}', \boldsymbol{y}', \boldsymbol{A}', \boldsymbol{P}')$ which describes the same code. Thus, without loss of generality, after possibly replacing $a$ by $ay_j$ and $b$ by $by_j$, one can suppose that $y_j = 1$. Moreover, after possibly replacing $\psi_j$ by $d\psi_j$, one can suppose that $d = 1$. Recall that in this section we suppose that $cd \neq 0$.

Thanks to these simplifying choices, (20) becomes

$$\begin{array}{rcl} c_i & = & a f(x_j) + c\psi_j(f) \\ c_{\tau(i)} & = & b f(x_j) + \psi_j(f). \end{array}$$

### 5.4.2 Shortening $\mathscr{C}^{(i)}$ at the last position to recover $x_j$

If we shorten $\mathscr{C}^{(i)}$ at the $\tau(i)$-th position, according to Lemma 24, it will "derandomise" the $i$-th position (it implies $\psi_j(f) = -bf(x_j)$) and any $\boldsymbol{c} \in \mathcal{S}_{\{\tau(i)\}}\left(\mathscr{C}^{(i)}\right)$ verifies

$$c_i = (a - bc) f(x_j).$$

Since the support $x_j$ and multiplier $y_j$ are known at all the positions of $\mathscr{C}^{(i)}$ but the two PR ones, for any codeword $\boldsymbol{c} \in \mathcal{S}_{\{\tau(i)\}}\left(\mathscr{C}^{(i)}\right)$, one can find the polynomial $f \in \mathbb{F}_q[x]_{<k}$ whose evaluation provides $\boldsymbol{c}$. Therefore, by collecting a basis of codewords in $\mathcal{S}_{\{\tau(i)\}}\left(\mathscr{C}^{(i)}\right)$ and the corresponding polynomials, we can recover the values of $x_j$ and $a - bc$.

### 5.4.3 Recovering the $2 \times 2$ matrix

Once we have $x_j$ we need to recover the matrix

$$\boldsymbol{A} = \begin{pmatrix} a & b \\ c & 1 \end{pmatrix}.$$

Note that, its determinant $\det \boldsymbol{A} = a - bc$ has already been obtained in the previous section. First, one can guess $b$ as follows. Let $\boldsymbol{G}^{(i)}$ be a generator matrix of $\mathscr{C}^{(i)}$. As in the previous section, by interpolation, one can compute the polynomials $f_1, \ldots, f_k$ whose evaluations provide the rows of $\boldsymbol{G}^{(i)}$. Consider the column vector

$$\boldsymbol{v} \stackrel{\text{def}}{=} \begin{pmatrix} f_1(x_j) \\ \vdots \\ f_k(x_j) \end{pmatrix}$$

and denote by $\boldsymbol{v}_i$ and $\boldsymbol{v}_{\tau(i)}$ the columns of $\boldsymbol{G}^{(i)}$ corresponding to positions $c_i$ and $c_{\tau(i)}$:

$$\boldsymbol{v}_i = \begin{pmatrix} af_1(x_j) + c\psi_j(f_1) \\ \vdots \\ af_k(x_j) + c\psi_j(f_k) \end{pmatrix} \qquad \text{and} \qquad \boldsymbol{v}_{\tau(i)} = \begin{pmatrix} bf_1(x_j) + \psi_j(f_1) \\ \vdots \\ bf_k(x_j) + \psi_j(f_k) \end{pmatrix}.$$

Next, search $\lambda \in \mathbb{F}_q$ such that $\boldsymbol{v}_i - \lambda \boldsymbol{v}_{\tau(i)}$ is collinear to $\boldsymbol{v}$. This relation of collinearity can be expressed in terms of cancellation of some $2 \times 2$ determinants which are polynomials of degree 1 in $\lambda$. Their common root is nothing but $c$.

Finally, we can find the pair $(a, b)$ by searching the pairs $(\lambda, \mu)$ such that

(i) $\lambda - c\mu = \det \boldsymbol{A}$;

(ii) $\boldsymbol{v}_i - \lambda \boldsymbol{v}$ and $\boldsymbol{v}_{\tau(i)} - \mu \boldsymbol{v}$ are collinear.

Here the relation of collinearity will be expressed as the cancellation of $2 \times 2$ determinants which are linear combinations of $\lambda, \mu$ and $\lambda\mu$ and elementary elimination process provides us with the value of the pair $(a, b)$.

## 6 Complexity of the attack

The most expensive part of the attack is the step consisting in identifying pairs of twin positions. Recall that, from [8], the computation of the square of a code of length $n$ and dimension $k$ costs $O(k^2 n^2)$ operations in $\mathbb{F}_q$. We need to compute the square of a code $O(w)$ times, because there are $w$ pairs of twin positions. Hence this step has a total complexity of $O(wn^2k^2)$ operations in $\mathbb{F}_q$. Note that the actual dimension of the shortened codes is significantly less than $k$ and hence the previous estimate is overestimated.

The cost of the Sidelnikov Shestakov attack is that of a Gaussian elimination, namely $O(nk^2)$ operations in $\mathbb{F}_q$ which is negligible compared to the previous step.

The cost of the final part is also negligible compared to the computation of the squares of shortened codes. This provides an overall complexity in $O(wn^2k^2)$ operations in $\mathbb{F}_q$.

## Conclusion

We presented a polynomial time key-recovery attack based on a square code distinguisher against the public key encryption scheme RLCE. This attack allows us to break all the so-called *odd ID* parameters suggested in [17]. Namely, the attack breaks the parameter sets for which the number $w$ of random columns was strictly less than $n - k$. Our analysis suggests that, for this kind of distinguisher by squaring shortenings of the code, the case $w = n - k$ is the critical one. The *even ID* parameters of [17], for which the relation $w = n - k$ always holds, remain out of the reach of our attack.

## Acknowledgements

The authors are supported by French *Agence nationale de la recherche* grants ANR-15-CE39-0013-01 *Manta* and ANR-17-CE39-0007 *CBCrypt*. Computer aided calculations have been performed using softwares Sage and Magma [5].

## References


[1] M. Baldi, M. Bianchi, F. Chiaraluce, J. Rosenthal, and D. Schipani. Enhanced public key security for the McEliece cryptosystem. *J. Cryptology*, 29(1):1–27, 2016.

[2] M. Bardet, J. Chaulet, V. Dragoi, A. Otmani, and J.-P. Tillich. Cryptanalysis of the McEliece public key cryptosystem based on polar codes. In *Post-Quantum Cryptography 2016*, LNCS, pages 118–143, Fukuoka, Japan, Feb. 2016.

[3] T. P. Berger and P. Loidreau. Security of the Niederreiter form of the GPT public-key cryptosystem. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT 2002*, page 267. IEEE, June 2002.

[4] D. J. Bernstein, T. Chou, T. Lange, I. von Maurich, R. Niederhagen, E. Persichetti, C. Peters, P. Schwabe, N. Sendrier, J. Szefer, and W. Wen. Classic McEliece: conservative code-based cryptography. `https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/round-1/submissions/Classic_McEliece.zip`, Nov. 2017. First round submission to the NIST post-quantum cryptography call.

[5] W. Bosma, J. Cannon, and C. Playoust. The Magma algebra system I: The user language. *J. Symbolic Comput.*, 24(3/4):235–265, 1997.

[6] I. Cascudo, R. Cramer, D. Mirandola, and G. Zémor. Squares of random linear codes. *IEEE Trans. Inform. Theory*, 61(3):1159–1173, March 2015.

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

[8] A. Couvreur, P. Gaborit, V. Gauthier-Umaña, A. Otmani, and J.-P. Tillich. Distinguisher-based attacks on public-key cryptosystems using Reed-Solomon codes. *Des. Codes Cryptogr.*, 73(2):641–666, 2014.

[9] A. Couvreur, I. Márquez-Corbella, and R. Pellikaan. Cryptanalysis of McEliece cryptosystem based on algebraic geometry codes and their subcodes. *IEEE Trans. Inform. Theory*, 63(8):5404–5418, Aug 2017.

[10] A. Couvreur, A. Otmani, and J.-P. Tillich. Polynomial time attack on wild McEliece over quadratic extensions. *IEEE Trans. Inform. Theory*, 63(1):404–427, Jan 2017.

[11] A. Couvreur, A. Otmani, J.-P. Tillich, and V. 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] J.-C. Faugère, V. Gauthier, A. Otmani, L. Perret, and J.-P. Tillich. A distinguisher for high rate McEliece cryptosystems. *IEEE Trans. Inform. Theory*, 59(10):6830–6844, Oct. 2013.

[13] W. C. Huffman and V. Pless. *Fundamentals of error-correcting codes*. Cambridge University Press, Cambridge, 2003.

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

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

[16] Y. Wang. Quantum resistant random linear code based public key encryption scheme RLCE. In *Proc. IEEE Int. Symposium Inf. Theory - ISIT 2016*, pages 2519–2523, Barcelona, Spain, July 2016. IEEE.

[17] Y. Wang. RLCE–KEM. `http://quantumca.org`, 2017. First round submission to the NIST post-quantum cryptography call.

[18] C. 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.

[19] C. Wieschebrink. Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. In *Post-Quantum Cryptography 2010*, volume 6061 of *LNCS*, pages 61–72. Springer, 2010.

# A How to treat the case of degenerate twin positions?

Recall that a pair of twin positions $i, \tau(i)$ is such that any codeword $\boldsymbol{c} \in \mathscr{C}$ has $i$–th and $\tau(i)$–th entries of he form:

$$\boldsymbol{c}_i = a y_j f(x_j) + b\psi_j(f) \qquad \boldsymbol{c}_{\tau(i)} = c y_j f(x_j) + d\psi_j(f).$$

This pair is said to be *degenerated* if either $b$ or $d$ is zero. In such a situation, some of the steps of the attack cannot be applied. In what follows, we explain how this rather rare issue can be addressed.

If either $b$ or $d$ is zero, then one of the positions is actually a pure GRS position while the other one is PR but the process explained in the article does not manage to associate the two twin columns.

Suppose w.l.o.g. that $b = 0$. In the first part if the attack, when we collect pairs of twin positions, the position $\tau(i)$ will be identified as PR with no twin sister *a priori*. To find its twin sister, we can proceed as follows. For any GRS position $j$ replace the $j$–th column $\boldsymbol{v}_j$ of a generator matrix $\boldsymbol{G}$ of $\mathscr{C}$ by an arbitrary linear combination of $\boldsymbol{v}_j$ and the $\tau(i)$–th column, this will "pseudo–randomise" this column and if the $j$–th column is the twin of the $\tau(i)$–th one, this will be detected by the process of shortening, squaring and searching zero columns in the parity check matrix.