



**HAL**  
open science

# An analysis of Coggia-Couvreur Attack on Loidreau's Rank-metric public-key encryption scheme in the general case

Pierre Loidreau, Ba-Duc Pham

► **To cite this version:**

Pierre Loidreau, Ba-Duc Pham. An analysis of Coggia-Couvreur Attack on Loidreau's Rank-metric public-key encryption scheme in the general case. 2022. hal-03514106

**HAL Id: hal-03514106**

**<https://hal.science/hal-03514106>**

Preprint submitted on 20 Jan 2022

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# An analysis of Coggia-Couvreur Attack on Loidreau's Rank-metric public-key encryption scheme in the general case

Pierre Loidreau<sup>1</sup> and Ba-Duc Pham<sup>2</sup>

<sup>1</sup> Univ Rennes, DGA MI, CNRS, IRMAR - UMR 6625, F-35000 Rennes, France  
 pierre.loidreau@univ-rennes1.fr

<sup>2</sup> Univ Rennes, IRMAR - UMR 6625, F-35000 Rennes, France  
 ba-duc.pham@univ-rennes1.fr

**Abstract.** In this paper we show that in the case where the public-key can be distinguished from a random code in Loidreau's encryption scheme, then Coggia-Couvreur attack can be extended to recover an equivalent secret key. This attack can be conducted in polynomial-time if the masking vector space has dimension 3, thus recovering the results of Ghatak.

**Keywords:** Rank metric codes, Gabidulin codes, code based cryptography, cryptanalysis

## Introduction

Since the use of  $\mathbb{F}_{q^m}$ -linear rank metric permits to design a short public key encryption scheme, one of the directions of code based cryptography consists in instantiating McEliece encryption scheme [1] with codes in rank metric, [2, 3].

Because of the structure of Gabidulin codes, any cryptosystem instantiated with codes containing Gabidulin codes not sufficiently scrambled was attacked [4]. In 2017, Loidreau proposed a scheme based on Gabidulin codes masked with a small dimensional vector space [5]. If the dimension of the vector space is too small, then there exists a very simple polynomial-time distinguishing algorithm.

The question was to know if distinguishing is enough to break. Coggia and Couvreur [6] showed that in the case where the dimension of the masking space is 2, a decryption procedure can be recovered in polynomial-time. More recently, Ghatak [7] claimed that this approach could be extended to a masking space of dimension 3.

In this work we show that this can be extended to any dimension. The attack is not necessary polynomial, but we include the previous results. Moreover we are able to prove rigorously under some assumptions the efficiency of the attack.

## Contribution

1. Completing the key-recovery attack for any  $\lambda$
2. In [7], the author uses one reduced polynomial to determine  $\gamma = \{\gamma_1, \gamma_2\}$  where  $\gamma$  specifies the secret subspace but their assumption is not clear in practice when we implement in MAGMA. In this paper, the use of system of polynomial equation gives a proof for the equivalent between the set of roots and the orbit of one root under the action of  $\mathbf{PGL}(3, \mathbb{F}_q)$ . It completes the polynomial time key recovery attack.

**Organization of the article** The first section outlines Loidreau’s scheme. In the beginning of the next section, we formulate the distinguisher for any dimension of the secret subspace. Afterwards, we describe the attack for any  $\lambda$  in 4 steps. In the end of this section, we analyse the complexity of the attack in case  $\lambda = 3$ . We conclude with a discussion on the results and future works.

## 1 The encryption scheme

### 1.1 Generalities

Let  $\mathbf{G}$  a random generator matrix of a Gabidulin code  $\mathcal{G}_k(\mathbf{g})$ . Fix an integer  $\lambda \leq m$  and an  $\mathbb{F}_q$ -vector subspace  $\mathcal{V}$  of  $\mathbb{F}_{q^m}$  of dimension  $\lambda$ . Let  $\mathbf{P} \in \mathbf{GL}(n, \mathbb{F}_{q^m})$  whose entries are all in  $\mathcal{V}$ . Then, let

$$\mathbf{G}_{\text{pub}} = \mathbf{G}\mathbf{P}^{-1}$$

- KeyGen: Public key  $(\mathbf{G}_{\text{pub}}, t)$  where  $t = \lfloor \frac{n-k}{2\lambda} \rfloor$   
Secret key  $(\mathbf{g}, \mathbf{P})$
- Encryption: Given a plaintext  $\mathbf{m} \in \mathbb{F}_{q^k}$ , choose  $\mathbf{e} \in \mathbb{F}_{q^m}^n$  of rank weight  $t$ . The ciphertext is:

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

- Decryption:
  - Compute  $\mathbf{c}\mathbf{P} = \mathbf{m}\mathbf{G} + \mathbf{e}\mathbf{P}$ .
  - Decode in  $\mathcal{G}_k(\mathbf{g})$  and  $\text{rk}(\mathbf{e}\mathbf{P}) \leq t\lambda \leq \frac{n-k}{2}$

Let us denote by  $\mathcal{C}_{\text{pub}}$  the code generated by  $\mathbf{G}_{\text{pub}}$  and by  $\mathcal{C}_{\text{pub}}^\perp$ , the dual code. Let  $\mathbf{H}_{\text{pub}}$  be a generator matrix of  $\mathcal{C}_{\text{pub}}^\perp$ . It is immediate that

$$\mathbf{H}_{\text{pub}} = \mathbf{H}_{\text{sec}}\mathbf{P}^T$$

where  $\mathbf{H}_{\text{sec}}$  is a parity-check matrix of  $\mathcal{G}_k(\mathbf{g})$ .

## 1.2 Goal of a reconstructing attack and solution set

Our main goal is to design a reconstructing attack from the knowledge of  $\mathcal{C}_{\text{pub}}^\perp$  and under some particular sets of parameters.

W.l.o.g, one can suppose that  $1 \in \mathcal{V}$ . Suppose that  $\mathcal{V} = \langle 1, \beta_1, \dots, \beta_{\lambda-1} \rangle_{\mathbb{F}_q}$  for some  $\{\beta_i\}_{i=1}^{\lambda-1} \in \mathbb{F}_{q^m} \setminus \mathbb{F}_q$ . Therefore,  $\mathbf{P}^T$  can be decomposed into

$$\mathbf{P}^T = \mathbf{P}_0 + \sum_{i=1}^{\lambda-1} \beta_i \mathbf{P}_i$$

where  $\mathbf{P}_i$  are  $n \times n$  matrices with entries in  $\mathbb{F}_q$  not necessarily invertible.

Let  $\mathcal{C}_{\text{sec}}^\perp$  the dual code of  $\mathcal{G}_k(\mathbf{g})$ . Thus,  $\mathcal{C}_{\text{sec}}^\perp = \mathcal{G}_{n-k}(\mathbf{a})$  for some  $\mathbf{a} \in \mathbb{F}_{q^m}^n$  with  $\text{rk}(\mathbf{a}) = n$ . We define

$$\mathbf{h}_0 = \mathbf{a}\mathbf{P}_0, \mathbf{h}_1 = \mathbf{a}\mathbf{P}_1, \dots, \mathbf{h}_{\lambda-1} = \mathbf{a}\mathbf{P}_{\lambda-1}$$

**Lemma 1.** *The code  $\mathcal{C}_{\text{pub}}^\perp$  is spanned by  $\mathbf{h}_0^{[i]} + \sum_{j=1}^{\lambda-1} \beta_j \mathbf{h}_j^{[i]}$  for  $i = 0, \dots, n-k-1$*

*Proof.* For any  $\mathbf{c} \in \mathcal{C}_{\text{pub}}^\perp$ , there exists  $P \in \mathbb{F}_{q^m}[X; \theta]$  of degree smaller than  $n-k$  such that

$$\mathbf{c} = P\langle \mathbf{a} \rangle \mathbf{P}^T = P\langle \mathbf{a} \rangle \mathbf{P}_0 + \sum_{i=1}^{\lambda-1} \beta_i P\langle \mathbf{a} \rangle \mathbf{P}_i = P\langle \mathbf{h}_0 \rangle + \sum_{i=1}^{\lambda-1} \beta_i P\langle \mathbf{h}_i \rangle$$

□

Let us define the so-called solution set of the encryption scheme

**Definition 1 (Solution set).** *The set  $\mathcal{S}$  of all  $(\mathbf{h}, \vec{\beta}) \in (\mathbb{F}_{q^m}^n)^\lambda \times \mathbb{F}_{q^m}^{\lambda-1}$  such that*

$$\mathcal{C}_{\text{pub}}^\perp = \left\langle \mathbf{h}_0^{[i]} + \sum_{j=1}^{\lambda-1} \beta_j \mathbf{h}_j^{[i]}, i = 0, \dots, n-k-1 \right\rangle \quad (1)$$

where  $\forall j = 0, \dots, \lambda$ ,  $\mathbf{h}_j$  has rank  $n$  and  $\langle 1, \beta_1, \dots, \beta_{\lambda-1} \rangle_{\mathbb{F}_q}$  has dimension  $\lambda$  is called solution set of the encryption scheme.

It is obvious that finding an element of the solution set  $\mathcal{S}$  implies the ability to design a polynomial-time decryption algorithm. What we call a reconstructing attack corresponds to finding an element in  $\mathcal{S}$ . The solution set  $\mathcal{S}$  has the following properties.

**Proposition 1.** *Let  $(\mathbf{h}, \vec{\beta}) \in (\mathbb{F}_{q^m}^n)^\lambda \times \mathbb{F}_{q^m}^{\lambda-1}$ . Let  $\mathbf{A} = (a_{j,i})_{j,i=0}^{\lambda-1} \in GL_\lambda(\mathbb{F}_q)$ . Let us define the following group action on  $(\mathbb{F}_{q^m}^n)^\lambda \times \mathbb{F}_{q^m}^{\lambda-1}$  by  $\mathbf{A} \cdot (\mathbf{h}, \vec{\beta}) = (\mathbf{h}', \vec{\beta}')$*

where

$$\left\{ \begin{array}{l} \mathbf{h}_j = \frac{a_{j,0}\mathbf{h}'_0 + \sum_{i=1}^{\lambda-1} a_{j,i}\mathbf{h}'_i}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0}\beta_i}, \quad j = 0, \dots, \lambda-1 \\ \beta'_j = \frac{a_{0,j} + \sum_{i=1}^{\lambda-1} a_{i,j}\beta_i}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0}\beta_i}, \quad j = 1, \dots, \lambda-1 \end{array} \right.$$

1. Then if  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$  we have  $\mathbf{A} \cdot (\mathbf{h}, \vec{\beta}) \in \mathcal{S}$
2. Moreover let  $\overline{\mathbf{A}} = \{\mathbf{B} \in GL_\lambda(\mathbb{F}_q) \mid \exists c \in \mathbb{F}_q^*, \mathbf{B} = c\mathbf{A}\}$ . Then, for any  $\mathbf{B} \in \overline{\mathbf{A}}$ , and for any  $(\mathbf{h}, \vec{\beta}) \in (\mathbb{F}_{q^m}^n)^\lambda \times \mathbb{F}_{q^m}^{\lambda-1}$  we have

$$\mathbf{A} \cdot (\mathbf{h}, \vec{\beta}) = \mathbf{B} \cdot (\mathbf{h}, \vec{\beta})$$

*Proof.* Let  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$ . Since from the definition of  $\mathcal{S}$  the elements  $1, \beta_1, \dots, \beta_{\lambda-1}$  are  $\mathbb{F}_q$ -linearly independent, and since  $\mathbf{A}$  is non singular,  $(a_{0,0}, \dots, a_{0,\lambda-1}) \neq 0$  this implies that  $a_{0,0} + \sum_{i=1}^{\lambda-1} a_{0,j}\beta_i \neq 0$ . Therefore, the elements  $\beta'_1, \dots, \beta'_{\lambda-1}$  are well defined and for all  $0 \leq \ell \leq n - k - 1$ ,

$$\begin{aligned} & \mathbf{h}'_0^{[\ell]} + \sum_{j=1}^{\lambda-1} \beta'_j \mathbf{h}'_j^{[\ell]} \\ &= \frac{1}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0}\beta_i} \left( \left( a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0}\beta_i \right) \mathbf{h}'_0^{[\ell]} + \sum_{j=1}^{\lambda-1} \left( a_{0,j} + \sum_{i=1}^{\lambda-1} a_{i,j}\beta_i \right) \mathbf{h}'_j^{[\ell]} \right) \\ &= \frac{1}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0}\beta_i} \left( \left( a_{0,0}\mathbf{h}'_0 + \sum_{j=1}^{\lambda-1} a_{0,j}\mathbf{h}'_j \right)^{[\ell]} + \sum_{i=1}^{\lambda-1} \beta_i \left( a_{i,0}\mathbf{h}'_0 + \sum_{j=1}^{\lambda-1} a_{i,j}\mathbf{h}'_j \right)^{[\ell]} \right) \\ &= \left( a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0}\beta_i \right)^{[\ell]-1} \left( \mathbf{h}'_0^{[\ell]} + \sum_{i=1}^{\lambda-1} \beta_i \mathbf{h}'_i^{[\ell]} \right) \end{aligned}$$

Therefore

$$\mathcal{C}_{\text{pub}}^\perp \stackrel{\text{def}}{=} \left\langle \mathbf{h}'_0^{[\ell]} + \sum_{i=1}^{\lambda-1} \beta_i \mathbf{h}'_i^{[\ell]}, 0 \leq \ell \leq n - k - 1 \right\rangle = \left\langle \mathbf{h}'_0^{[\ell]} + \sum_{i=1}^{\lambda-1} \beta'_i \mathbf{h}'_i^{[\ell]}, 0 \leq \ell \leq n - k - 1 \right\rangle$$

Thus,  $(\mathbf{h}', \vec{\beta}') \in \mathcal{S}$ .

For the second point of the proposition : Let  $\mathbf{B} \in \overline{\mathbf{A}}$ . It means that there exists  $c \in \mathbb{F}_q^*$  such that  $\mathbf{B}\mathbf{A}^{-1} = c\mathbf{I}$  or equivalently  $\mathbf{B} = c\mathbf{A}$ .

Let  $(\mathbf{h}'', \vec{\beta}'') \stackrel{def}{=} \mathbf{B} \cdot (\mathbf{h}', \vec{\beta}')$ , where  $(\mathbf{h}', \vec{\beta}') = \mathbf{A} \cdot (\mathbf{h}, \vec{\beta}) \in (\mathbb{F}_{q^m}^n)^\lambda \times \mathbb{F}_{q^m}^{\lambda-1}$ . It means that

$$\left\{ \begin{array}{l} \mathbf{h}'_j = \frac{b_{j,0} \mathbf{h}_0'' + \sum_{i=1}^{\lambda-1} b_{j,i} \mathbf{h}_i''}{b_{0,0} + \sum_{i=1}^{\lambda-1} b_{i,0} \beta_i}, \quad j = 0, \dots, \lambda - 1 \\ \beta_i'' = \frac{b_{0,j} + \sum_{i=1}^{\lambda-1} b_{i,j} \beta_i}{b_{0,0} + \sum_{i=1}^{\lambda-1} b_{i,0} \beta_i}, \quad j = 1, \dots, \lambda - 1 \end{array} \right.$$

It is obvious that the actions of 2 matrix  $\mathbf{A}, \mathbf{B}$  give us same image, which is  $(\mathbf{h}'', \vec{\beta}'') = (\mathbf{h}', \vec{\beta}')$ . □

## 2 Attacks on the system

Now the section is organised as follows: In a first part we make a brief summary of the attack. Since it is technical, this section highlights the different principles. In first subsection, we introduce the distinguisher between the dual of the public code and the random codes. Afterwards, we make some assumptions in the beginning and under these assumptions, we exploit the attack based on the distinguisher introduced in 2.1. We also consider the special case of  $\lambda = 3$  and analyze its complexity in Section 2.3.

We will also need to introduce a special setting to simplify the technicalities of the proofs. For any  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$ , we denote

$$\mathbf{y}_{(\mathbf{h}, \vec{\beta})}^{[u,j]} = \mathbf{h}_0^{[j]} + \sum_{i=1}^{\lambda-1} \beta_i^{[u]} \mathbf{h}_i^{[j]}$$

for any integers  $(u, j)$ . For a given  $(\mathbf{h}, \vec{\beta})$ , to simplify, we denote it by  $\mathbf{y}^{[u,j]}$ . Moreover, we can denote  $\mathbf{y}^{[M]} = \{\mathbf{y}^{[u,j]}, (u, j) \in M \subset \mathbb{Z} \times \mathbb{Z}\}$ . The codes that we will consider will be generated by the  $\mathbf{y}^{[u,j]}$ , where  $(u, j) \in \mathbb{Z} \times \mathbb{Z}$ . We denote by

$$\mathcal{C}_I \stackrel{def}{=} \langle \mathbf{y}^{[u,j]}, (u, j) \in I \rangle,$$

if  $I$  is non-empty and  $\mathcal{C}_\emptyset \stackrel{def}{=} \{0\}$ . From the expression of  $\mathcal{C}_{\text{pub}}^\perp$  in Definition 1, we have

$$\mathcal{C}_{\text{pub}}^\perp = \mathcal{C}_{\{0\} \times [0, \dots, n-k-1]} \quad (2)$$

Now we introduce a very fundamental theorem which will support all of our future proofs

**Theorem 1 (CodeSet theorem).** *We have*

$$\forall I, J \subset \mathbb{Z} \times \mathbb{Z}, \begin{cases} \mathcal{C}_{I \cup J} = \mathcal{C}_I + \mathcal{C}_J \\ \mathcal{C}_{I \cap J} \subset \mathcal{C}_I \cap \mathcal{C}_J \end{cases}$$

If moreover  $M \subset \mathbb{Z} \times \mathbb{Z}$ , where  $\mathbf{y}^{[u,j]}$ ,  $(u,j) \in M$  are  $\mathbb{F}_q^m$  linearly independent, then

- for all  $I \subset M$ ,  $\dim(\mathcal{C}_I) = |I|$
- $\forall I, J \subset M$ ,  $\mathcal{C}_I \cap \mathcal{C}_J = \mathcal{C}_{I \cap J}$
- $\forall I, J \subset M$ ,  $\mathcal{C}_{I \sqcup J} = \mathcal{C}_I \oplus \mathcal{C}_J$ , where  $\sqcup$  means that the sets do not intersect.

*Proof.* The code generating set for  $\mathcal{C}_I + \mathcal{C}_J$  is the union of generating sets for  $\mathcal{C}_I$  and  $\mathcal{C}_J$ , since  $\mathbf{y}^{[I]}$  and  $\mathbf{y}^{[J]}$  are generating sets respectively for  $\mathcal{C}_I$  and  $\mathcal{C}_J$  then  $\mathbf{y}^{[I \cup J]}$  is a generating set for  $\mathcal{C}_I + \mathcal{C}_J$ . Hence  $\mathcal{C}_{I \cup J} = \mathcal{C}_I + \mathcal{C}_J$ . Now the generating set  $\mathbf{y}^{[I \cap J]}$  of  $\mathcal{C}_{I \cap J}$  is included in the generating set of  $\mathcal{C}_I$  and of  $\mathcal{C}_J$ . Therefore  $\mathcal{C}_{I \cap J} \subset \mathcal{C}_I \cap \mathcal{C}_J$ .

Let us consider now  $M$  such that  $\mathbf{y}^{[M]}$  is formed with linearly independent vectors. It is immediate that for any  $I \subset M$ , a basis of  $\mathcal{C}_I$  is  $\mathbf{y}^{[I]}$ , therefore the dimension of  $\mathcal{C}_I$  is exactly equal to  $|I|$ . Let  $\mathbf{c} \in \mathcal{C}_I \cap \mathcal{C}_J$ . We have

$$\mathbf{c} = \sum_{(u,j) \in I} c_{u,j} \mathbf{y}^{[u,j]} = \sum_{(u,j) \in I \setminus J} c_{u,j} \mathbf{y}^{[u,j]} + \sum_{(u,j) \in I \cap J} c_{u,j} \mathbf{y}^{[u,j]}$$

and similarly

$$\mathbf{c} = \sum_{(u,j) \in J} c'_{u,j} \mathbf{y}^{[u,j]} = \sum_{(u,j) \in J \setminus I} c'_{u,j} \mathbf{y}^{[u,j]} + \sum_{(u,j) \in I \cap J} c'_{u,j} \mathbf{y}^{[u,j]}$$

Since by hypotheses on  $M$ , the  $\mathbf{y}^{[M]}$  are linearly independent, this implies the equality of the coefficients on this bases and thus that  $c_{u,j} = 0$ , for  $(u,j) \in I \setminus J$ . Therefore,  $\mathbf{c} \in \mathcal{C}_{I \cap J}$ .

The last item comes from the fact that is  $I$  and  $J$  do not intersect then  $I \cap J = \emptyset$ , thus  $\mathcal{C}_I \cap \mathcal{C}_J = \{0\}$ . □

## 2.1 A distinguishing attack in the general case

We show that if  $n, k, \lambda$  satisfy  $k > \frac{(\lambda-1)n}{\lambda} + 1$ , then one can distinguish the public-code from a random code in polynomial time. First we prove the following theorem.

**Theorem 2.**  $\dim_{\mathbb{F}_q^m} \left( \mathcal{C}_{pub}^\perp + \mathcal{C}_{pub}^{\perp [1]} + \dots + \mathcal{C}_{pub}^{\perp [\lambda]} \right) \leq \lambda(n - k) + \lambda$

*Proof.* Let  $\mathcal{S}_0 \stackrel{\text{def}}{=} \sum_{i=0}^{\lambda-1} \mathcal{C}_{\text{pub}}^{\perp [i]}$ . We want to show that  $\dim(\mathcal{S}_0 + \mathcal{C}_{\text{pub}}^{\perp [\lambda]}) \leq \lambda(n - k) + \lambda$ . For any  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$  from the expression of  $\mathcal{C}_{\text{pub}}^{\perp}$  under the form (2) and from the CodeSet theorem we obtain

$$\mathcal{S}_0 = \mathcal{C}_{S_0}, \text{ where } S_0 = \bigsqcup_{u=0}^{\lambda-1} \{u\} \times [u, n - k + u - 1]$$

and

$$\mathcal{C}_{\text{pub}}^{\perp [\lambda]} = \mathcal{C}_{\{\lambda\} \times [\lambda, n - k + \lambda - 1]}$$

Let

$$I = \bigsqcup_{u=0}^{\lambda-1} \{u\} \times [\lambda - 1, n - k - 1] = [0, \lambda - 1] \times [\lambda - 1, n - k - 1]$$

We have clearly  $I \subset S_0$ , implying  $\mathcal{C}_I \subset \mathcal{S}_0$ .

By the hypotheses on  $\mathcal{S}$  the  $(1, \beta_1, \dots, \beta_{\lambda-1})$  are linearly independent over  $\mathbb{F}_q$ . Thus,

$$\det \begin{bmatrix} 1 & \beta_1 & \dots & \beta_{\lambda-1} \\ 1 & \beta_1^{[1]} & \dots & \beta_{\lambda-1}^{[1]} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & \beta_1^{[\lambda-1]} & \dots & \beta_{\lambda-1}^{[\lambda-1]} \end{bmatrix} \neq 0,$$

it implies that for any  $j \in \lambda - 1, \dots, n - k - 1$ ,  $\mathcal{C}_{[0, \lambda - 1] \times \{j\}} = \langle \mathbf{h}_i^{[j]}, 0 \leq i \leq \lambda - 1 \rangle$ . Hence,

$$\mathcal{C}_I = \left\langle \mathbf{h}_i^{[j]}, \begin{array}{l} 0 \leq i \leq \lambda - 1 \\ \lambda - 1 \leq j \leq n - k - 1 \end{array} \right\rangle$$

In particular from the structure of  $\mathbf{y}^{[u, j]}$  for any  $J \subset * \times [\lambda - 1, n - k - 1]$ , we have  $\mathcal{C}_J \subset \mathcal{C}_I \subset \mathcal{S}_0$ .

Thus,  $\mathcal{C}_{\{\lambda\} \times [\lambda, n - k - 1]} \subset \mathcal{S}_0 \cap \mathcal{C}_{\text{pub}}^{\perp [\lambda]}$ . From its construction,  $\mathcal{C}_{\text{pub}}^{\perp [\lambda]}$  has dimension  $n - k$ . The vectors  $\mathbf{y}^{[\lambda, j]}, j \in [\lambda, n - k + \lambda - 1]$  are linearly independent and from the CodeSet theorem,  $\mathcal{C}_{\{\lambda\} \times [\lambda, n - k - 1]}$  has dimension  $n - k - \lambda$ . Therefore,  $\dim(\mathcal{S}_0 \cap \mathcal{C}_{\text{pub}}^{\perp [\lambda]}) \geq n - k - \lambda$ . Conversely,

$$\begin{aligned} \dim(\mathcal{S}_0 + \mathcal{C}_{\text{pub}}^{\perp [\lambda]}) &= \dim(\mathcal{S}_0) + \dim(\mathcal{C}_{\text{pub}}^{\perp [\lambda]}) - \dim(\mathcal{S}_0 \cap \mathcal{C}_{\text{pub}}^{\perp [\lambda]}) \\ &\leq \lambda(n - k) + (n - k) - (n - k - \lambda) = \lambda(n - k) + \lambda \end{aligned}$$

□

Now the distinguishing attack comes from this proposition



**Proposition 2 ( [6] Proposition 2 ).** *If  $\mathcal{C}_{rand}$  is a random code of length  $n$  and dimension  $k$ , then for a non-negative integer  $a$  and a positive  $\lambda < k$ , we have*

$$\mathbb{P} \left( \dim_{\mathbb{F}_{q^m}} \left( \mathcal{C}_{rand} + \mathcal{C}_{rand}^{[1]} + \cdots + \mathcal{C}_{rand}^{[\lambda]} \right) \leq \min(n, (\lambda + 1)k) - a \right) = O(q^{-ma}).$$

Now whenever  $k > \frac{(\lambda-1)n}{\lambda} + 1$ , the dimension of  $\mathcal{C}_{rand} + \mathcal{C}_{rand}^{[1]} + \cdots + \mathcal{C}_{rand}^{[\lambda]}$  is very probably equal to  $(\lambda + 1)(n - k)$  whereas the dimension of  $\mathcal{C}_{pub}^\perp + \mathcal{C}_{pub}^{\perp [1]} + \cdots + \mathcal{C}_{pub}^{\perp [\lambda]}$  is probably equal to  $\lambda(n - k + 1)$  (since  $\lambda(n - k + 1) < n$ ), which is strictly less than  $(\lambda + 1)(n - k)$ .

## 2.2 Reconstructing attack

We suppose that the public code has rate larger than  $(\lambda - 1)/\lambda$ , so that the distinguisher introduced in Section 2.1 works on it.

Although the attack we describe should work heuristically, to have rigorous proofs of work we need the following assumptions, which are not very constraining

- (1) There exists an element  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$  such that  $\forall i_1, \dots, i_\lambda \in \{1, \dots, n - k - 1\}$  distinct.

$$\det \begin{bmatrix} 1 & \beta_1^{[i_1]} & \beta_2^{[i_1]} & \cdots & \beta_{\lambda-1}^{[i_1]} \\ 1 & \beta_1^{[i_2]} & \beta_2^{[i_2]} & \cdots & \beta_{\lambda-1}^{[i_2]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \beta_1^{[i_\lambda]} & \beta_2^{[i_\lambda]} & \cdots & \beta_{\lambda-1}^{[i_\lambda]} \end{bmatrix} \neq 0,$$

- (2)  $\dim_{\mathbb{F}_{q^m}} \mathcal{C}_{pub}^\perp + \mathcal{C}_{pub}^{\perp [1]} + \mathcal{C}_{pub}^{\perp [2]} + \cdots + \mathcal{C}_{pub}^{\perp [\lambda]} = \lambda(n - k) + \lambda$   
(3) There is no  $\overline{\mathbf{A}} \in \mathbf{PGL}(\lambda, \mathbb{F}_q) \setminus \overline{\mathbf{I}}_\lambda$  and  $\mathbf{A} = (a_{ij})_{i,j=1}^\lambda$  that satisfies

$$\beta_j = \frac{a_{0,j} + \sum_{i=1}^{\lambda-1} a_{i,j} \beta_i}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} \beta_i}, \quad \forall j = 1, \dots, \lambda - 1$$

The first step of the attack is dedicated to finding one dimensional vector-spaces  $\mathcal{A}_i$  for  $i = 1, \dots, n - k - 1$ , such that any element  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$  satisfies:

$$\mathcal{A}_i = \left\langle \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-i]} \mathbf{h}_j \right\rangle$$

From the  $\mathcal{A}_i$ 's, one obtains a system of  $\lambda - 1$  multivariate polynomials which are of degree  $q^{\lambda+1} - q^i$  for  $i = 1, \dots, \lambda - 1$  satisfied by all the vectors  $\vec{\beta}$  such that  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$ . Under the above assumptions, we can also prove the stabilization of the set of solution  $\mathcal{S}$  under the action of  $\mathbf{PGL}(\lambda, \mathbb{F}_q)$  in the end of 3.2.2.

The complexities of the steps (by operations over  $\mathbb{F}_{q^m}$ ):

- Step 1. It costs  $O(n^3 \log q)$  operations for computing  $\mathcal{C}_{\text{pub}}^\perp{}^{[i]}$  and  $O(n^{\omega+1})$  for taking the intersection.
- Step 2. The principal complexity of this is finding the roots of the system of polynomials. In case of  $\lambda = 3$ , it can be done in polynomial time where the complexity is  $\tilde{O}(\tilde{d}^2 n \log q)$  for  $\tilde{d} = (q^4 - q)(q^4 - q^2)$ .  
Once such a root is found, the remaining of step 2 needs a finite number of linear systems solving which costs  $O(n^\omega)$ .

**First step: Recovering one-dimensional vector spaces** We now suppose that the three assumptions in section 1.2 are true we have the following theorem:

**Theorem 3.** *Let  $d := n - k - \lambda + 1$ . Under Assumptions (1), (2), (3),*

---

**Algorithm 1: Recovering 1-dimensional vector spaces**

---

**Input:**  $\mathcal{C}_{\text{pub}}^\perp$ ,  $\lambda \leq \frac{n}{n-k+1}$   
**Output:**  $\mathcal{A}_i$  for  $i = 0, \dots, n - k - 1$

- 1  $\mathcal{S}_0 \leftarrow \mathcal{C}_{\text{pub}}^\perp{}^{[0]} + \mathcal{C}_{\text{pub}}^\perp{}^{[1]} + \dots + \mathcal{C}_{\text{pub}}^\perp{}^{[\lambda-1]}$
- 2  $\mathcal{A} \leftarrow \left( \bigcap_{i=0}^d \mathcal{S}_0^{[i]} \right)^{[-d]}$
- 3  $\mathcal{D}_{\lambda-1} \leftarrow \mathcal{A}^{[\lambda-2]} \cap \mathcal{C}_{\text{pub}}^\perp{}^{[\lambda-1-d]}$  and  $\mathcal{B}_0 \leftarrow \mathcal{A} + \mathcal{D}_{\lambda-1}^{[1-\lambda]}$
- 4  $\mathcal{D}_0 \leftarrow \mathcal{B}_0 \cap \mathcal{C}_{\text{pub}}^\perp{}^{[-1]}$
- 5 **for**  $\ell \in 1, \dots, \lambda - 2$  **do**
- 6      $\mathcal{B}_\ell \leftarrow \mathcal{A} + \sum_{j=0}^{\ell-1} \mathcal{D}_j^{[\ell-j]},$
- 7      $\mathcal{D}_\ell \leftarrow \mathcal{B}_\ell \cap \mathcal{C}_{\text{pub}}^\perp{}^{[-1]}$
- 8  $\mathcal{H} \leftarrow \sum_{j=0}^{\lambda-1} \mathcal{C}_j^{[2-j-\lambda]}$
- 9 **for**  $i \in 0, \dots, n - k - 1$  **do**
- 10     Return  $\mathcal{A}_i \leftarrow \mathcal{H} \cap \mathcal{C}_{\text{pub}}^\perp{}^{[-i]}$

---

Algorithm (1) returns the 1-dimensional vector spaces

$$\mathcal{A}_i = \left\langle \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-i]} \mathbf{h}_j \right\rangle, \quad i = 0, \dots, n - k - 1$$

for any  $(\mathbf{h}, \vec{\beta}) \in \mathcal{S}$ .

*Proof.* For the proof we will thus make intensive use of the CodeSet theorem. First from assumption (2) and theorem 2, the set

$$M = \bigsqcup_{u=0}^{\lambda} \{u\} \times [u, n - k + u - 1] = S_0 \sqcup \{\lambda\} \times [\lambda, n - k + \lambda - 1]$$

with cardinality  $\lambda(n-k) + \lambda$ , is such that  $\mathbf{y}^{[M]}$  is formed of linearly independent vectors and  $\mathcal{C}_M = \sum_{i=0}^{\lambda} \mathcal{C}_{\text{pub}}^{\perp [i]}$ . This point is very important since this is the crucial point of the proof of the theorem.

**Line 1.** From theorem 2 we have

$$\mathcal{S}_0 = \mathcal{C}_{S_0}, \text{ where } S_0 = \bigsqcup_{u=0}^{\lambda-1} \{u\} \times [u, n-k+u-1]$$

We can write  $S_0$  under the form

$$S_0 = \underbrace{\left( \bigsqcup_{u=0}^{\lambda-2} \{u\} \times [u, \lambda-2] \right)}_{I_1} \sqcup \underbrace{[0, \lambda-1] \times [\lambda-1, n-k-1]}_{I_2} \sqcup \underbrace{\left( \bigsqcup_{u=0}^{\lambda-1} \{u\} \times [n-k, n-k+u-1] \right)}_{I_3}$$

Let  $I_4 = I_3 \sqcup \{\lambda\} \times [n-k, n-k+\lambda-1]$ . With these notations, we have

$S_0 = I_1 \sqcup I_2 \sqcup I_3 \subset I_1 \sqcup I_2 \sqcup I_4 = M$ . Since their cardinalities satisfy

$$|I_1| = \frac{\lambda(\lambda-1)}{2}, |I_2| = \lambda(n-k-\lambda+1), |I_3| = \frac{\lambda(\lambda-1)}{2} \text{ and } |I_4| = \lambda,$$

from theorem 2 the dimension of  $\mathcal{S}_0$  and  $\mathcal{C}_M$  is exactly  $\lambda(n-k)$  and  $\lambda(n-k)+\lambda$  respectively, and additionally under the CodeSet theorem,

$$\begin{aligned} \mathcal{S}_0 &= \mathcal{C}_{I_1} \oplus \mathcal{C}_{I_2} \oplus \mathcal{C}_{I_3} \\ \mathcal{C}_M &= \mathcal{C}_{I_1} \oplus \mathcal{C}_{I_2} \oplus \mathcal{C}_{I_4} \end{aligned}$$

The set  $I_2$  corresponds to the set denoted by  $I$  in the proof of theorem 2. We have

$$\mathcal{C}_{I_2} = \left\langle \mathbf{h}_i^{[j]}, \begin{array}{l} 0 \leq i \leq \lambda-1 \\ \lambda-1 \leq j \leq n-k-1 \end{array} \right\rangle$$

This property gives us the flexibility for the modification of the set  $M$  to obtain several sets of indexes  $M'$  such that  $\mathbf{y}^{[M']}$  is formed of linearly independent vectors. It can be done by the replacement of the set  $[0, \lambda-1]$  by the set  $A_j$  of  $\lambda$  elements corresponding to any  $j$ . We can see it precisely as the following lemma:

**Lemma 2.** For every set  $I_2' = \bigsqcup_{j=\lambda-1}^{n-k-1} A_j \times \{j\}$  where  $|A_j| = \lambda$ , then  $M' = I_1 \sqcup$

$I_2' \sqcup I_4$  satisfies

- $\mathcal{C}_M = \mathcal{C}_{M'}$ .
- $\mathbf{y}^{[M']}$  is formed of linearly independent vectors.

*Proof.*  $\mathcal{C}_{I_2'} = \left\langle \mathbf{h}_i^{[j]}, \begin{array}{l} 0 \leq i \leq \lambda-1 \\ \lambda-1 \leq j \leq n-k-1 \end{array} \right\rangle = \mathcal{C}_{I_2}$  (Assumption (1)). Moreover,

$|I_2'| = |I_2| = \lambda(n-k-\lambda+1)$ . Hence  $\mathcal{C}_{M'} = \mathcal{C}_{I_1} \oplus \mathcal{C}_{I_2'} \oplus \mathcal{C}_{I_4} = \mathcal{C}_M$  and  $\mathbf{y}^{[M']}$  is formed of linearly independent vectors.  $\square$

Through this section, to apply the CodeSet theorem, in the beginning of each step, we will define its set of indexes  $M'$  such that  $\mathbf{y}^{[M']}$  are linearly independent vectors and it contains the set of indexes of subspace that we want to compute the intersection. To be convenient, we will use some images where the red dot  $\bullet$  are indexes of some transformation of  $I_1$ , the blue square  $\blacksquare$  are indexes of some transformation of  $I_4$ , the green  $\times$  are indexes of some transformation of  $I_2$  and the black diamond  $\blacklozenge$  are indexes of  $\mathcal{C}_{\text{pub}}^\perp$ . On the other hand, the integer points which are inside the blue figures are indexes of linearly independent vectors. The left triangular covers all the points of the set of indexes  $I_1$ , the right triangular covers all the points of the set of indexes  $I_4$  and the rectangular covers all the points of the set of indexes which is flexible modification of  $I_2$ .

To be convenient, for  $I \subset \mathbb{Z} \times \mathbb{Z}$  and  $a \in \mathbb{Z}$ , we denote  $I + a = \{(u + a, j + a), (u, j) \in I\}$ . In the figures, we can consider  $I + a$  as the translation of  $I$  by the vector  $(a, a)$ . For example, in the figure 1, the set of red points shows  $I_1$  in the first image and  $I_1 + d + 1$  in the second ones.

**Line 2.** We show that  $\mathcal{A} = \mathcal{C}_{I_1 \sqcup I_3 + (\lambda - (n-k) - 1)}$ .

**Lemma 3.** Let  $\mathcal{S}_i \stackrel{\text{def}}{=} \mathcal{C}_{\text{pub}}^\perp [i] + \mathcal{C}_{\text{pub}}^\perp [i+1] + \dots + \mathcal{C}_{\text{pub}}^\perp [i+\lambda-1]$ . For any set  $*$  of  $\lambda$  distinct integers modulo  $m$  we have

$$\forall 0 \leq d \leq n - k - \lambda + 1, \bigcap_{i=0}^d \mathcal{S}_i = \mathcal{C}_{(I_1+d) \sqcup * \times [\lambda-1+d, n-k-1] \sqcup I_3}$$

*Proof.* We prove the theorem by induction. This lemma is true for  $d = 0$ . We suppose that it is true until  $0 \leq d \leq n - k - \lambda$ , then we need to prove that it must be true for  $d + 1$ .

Indeed,

$$\begin{aligned} \bigcap_{i=0}^{d+1} \mathcal{S}_i &= \mathcal{S}_0 \cap \left( \bigcap_{i=0}^d \mathcal{S}_i \right)^{[1]} \\ &= \mathcal{C}_{S_0} \cap \mathcal{C}_{(I_1+d+1) \cup * \times [\lambda+d, n-k] \cup (I_3+1)} \end{aligned}$$

Let  $M_1 = I_1 \sqcup I_4 \sqcup [d+1, \lambda+d] \times [\lambda-1, n-k-1]$

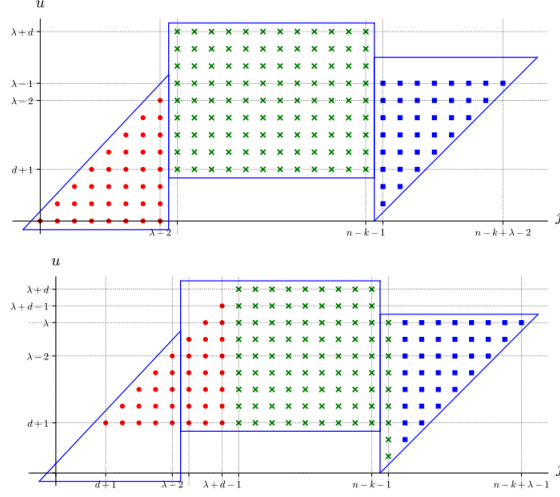
Concerning  $S_0 \subset M_1$  and  $J \stackrel{\text{def}}{=} (I_1+d+1) \cup * \times [\lambda+d, n-k] \cup (I_3+1) \subset M_1$ . We can apply the CodeSet theorem

$$\mathcal{C}_{S_0} \cap \mathcal{C}_J = \mathcal{C}_{S_0 \cap J}$$

And by a slightly fastidious computation on the sets intersections, we see that

$$S_0 \cap J = (I_1 + d + 1) \cup * \times [\lambda + d, n - k - 1] \cup I_3$$

It is not very difficult to check that the sets do not intersect which gives the result. □



**Fig. 1.** Points of  $\mathcal{S}_0$  (above) and  $(I_1 + d + 1) \cup * \times [\lambda + d, n - k] \cup (I_3 + 1)$

In the rest of the proof we will suppose that  $d = n - k - \lambda + 1$ . If we instantiate the lemma with  $d$  and elevate to the power  $[-d]$  we obtain the following corollary:

**Corollary 1.**  $\mathcal{A} = \mathcal{C}_{I_1 \sqcup I_3 - d}$

*Proof.* We have  $\bigcap_{i=0}^d \mathcal{S}_i = \mathcal{C}_{(I_1 + d) \sqcup I_3}$ , with  $I_1$  and  $I_3$  subsets of  $M$ . Thus, from CodeSet theorem we have

$$\mathcal{A}^{[d]} = \mathcal{C}_{I_1 + d} \oplus \mathcal{C}_{I_3}$$

Implying that  $\mathcal{A} = \mathcal{C}_{I_1} \oplus \mathcal{C}_{I_3 - d}$ . □

**Line 3.** Since  $\mathcal{C}_{\text{pub}}^\perp = \mathcal{C}_{\{0\} \times [0, n - k - 1]}$ , we have

$$\mathcal{A}^{[d-1]} \cap \mathcal{C}_{\text{pub}}^\perp = \mathcal{C}_{(I_1 + (d-1)) \sqcup (I_3 - 1)} \cap \mathcal{C}_{\{0\} \times [0, n - k - 1]}$$

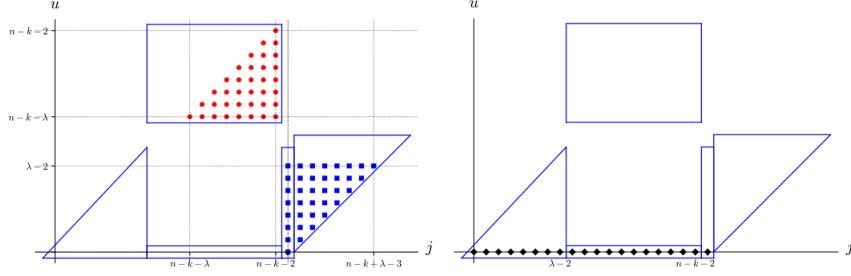
with

$$\begin{aligned} I_1 + (d - 1) &= \{(u, j) : d - 1 \leq u \leq j \leq n - k - 2\} \\ I_3 - 1 &= \{(u, j) : 0 \leq u \leq \lambda - 2, n - k - 1 \leq j \leq n - k + u - 1\} \end{aligned}$$

Let

$$I_2' = ((\{0\} \sqcup [n - k - \lambda, n - k - 2]) \times [\lambda - 1, n - k - 2]) \sqcup [0, \lambda - 1] \times \{n - k - 1\}$$

$$\text{and } M_2 = I_1 \sqcup I_4 \sqcup I_2'$$



**Fig. 2.** Points of  $(I_1 + (d - 1)) \sqcup (I_3 - 1)$  and  $\{0\} \times [0, n - k - 1]$

We can prove that  $I_1 + (d - 1)$ ,  $I_3 - 1$  and  $\{0\} \times [0, n - k - 1]$  are all in  $M_2$ . Now since  $\lambda \leq (n - k)/2$ , we have  $d \geq \lambda - 1$ , it implies that

$$((I_1 + (d - 1)) \sqcup (I_3 - 1)) \cap \{0\} \times [0, n - k - 1] = \{(0, n - k - 1)\}$$

Since the intersecting sets are all subsets of the set  $M_2$ , we can apply the CodeSet theorem, and we obtain

$$\mathcal{A}^{[d-1]} \cap \mathcal{C}_{\text{pub}}^\perp = \mathcal{C}_{\{(0, n-k-1)\}}$$

which by elevating to the power  $[\lambda - d - 1]$  gives

$$\mathcal{D}_{\lambda-1} = \mathcal{A}^{[\lambda-2]} \cap \mathcal{C}_{\text{pub}}^\perp^{[\lambda-d-1]} = \mathcal{C}_{\{(\lambda-d-1, 2\lambda-3)\}}$$

Note that elevating to the power  $[m]$  the scalars corresponds to the identity operator, we also have  $\mathcal{D}_{\lambda-1} = \mathcal{C}_{\{(m+\lambda-d-1, 2\lambda-3)\}}$ . This will be of use in the proof of the algorithm. Now since  $\mathcal{B}_0 = \mathcal{A} + \mathcal{D}_{\lambda-1}^{[1-\lambda]}$ , we deduce that

$$\mathcal{B}_0 = \mathcal{C}_{I_1 \sqcup I_3 - d} + \mathcal{C}_{\{(-d, \lambda-2)\}}$$

**Line 4.** Compute  $\mathcal{D}_0 = \mathcal{B}_0 \cap \mathcal{C}_{\text{pub}}^\perp^{[-1]}$

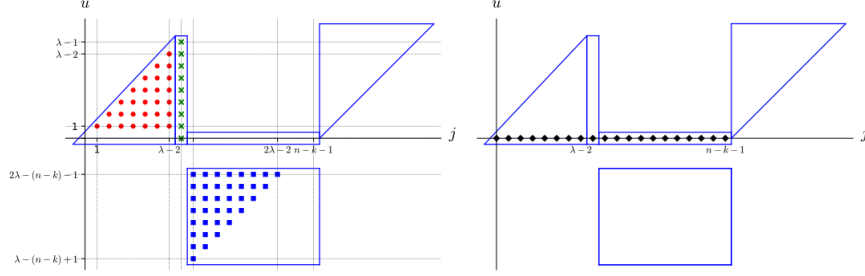
We compute

$$\begin{aligned} \mathcal{D}_0^{[1]} &= \mathcal{B}_0^{[1]} \cap \mathcal{C}_{\text{pub}}^\perp \\ &= (\mathcal{C}_{(I_1+1) \sqcup I_3 - (d-1)} + \mathcal{C}_{\{(-d+1, \lambda-1)\}}) \cap \mathcal{C}_{\{0\} \times [0, (n-k)-1]} \\ &= (\mathcal{C}_{(I_1+1) \setminus [1, \lambda-1] \times \{\lambda-1\}} \oplus \mathcal{C}_{*\times\{\lambda-1\}} \oplus \mathcal{C}_{I_3 - (d-1)}) \cap \mathcal{C}_{\{0\} \times [0, (n-k)-1]} \end{aligned}$$

where  $*$  is instantiated for the set  $[0, \lambda - 2] \sqcup \{-d + 1\}$  which contains  $\lambda$  different integers.

Let  $I_2' = ([0, \lambda - 1] \times \{\lambda - 1\}) \sqcup ([2 - d, \lambda - d] \times [\lambda, n - k - 1])$  and  $M_3 = I_1 \sqcup I_4 \sqcup I_2'$ . Now  $(I_1 + 1) \sqcup I_3 - (d - 1) \sqcup (-d + 1, \lambda - 1) \subset M_3$ . Moreover,

$$((I_1 + 1) \sqcup I_3 - (d - 1)) \cap \{0\} \times [0, (n - k) - 1] = \emptyset$$



**Fig. 3.** Points of  $((I_1 + 1) \setminus [1, \lambda - 1] \times \{\lambda - 1\}) \sqcup * \times \{\lambda - 1\} \sqcup I_3 - (d - 1)$  and  $\{0\} \times [0, (n - k) - 1]$

and  $* \times \{\lambda - 1\} \cap \{0\} \times [0, n - k - 1] = \{(0, \lambda - 1)\}$

Hence, by the CodeSet theorem,

$$\begin{aligned} \mathcal{B}_0^{[1]} \cap \mathcal{C}_{\text{pub}}^\perp &= \mathcal{C}_{\{(0, \lambda - 1)\}} \\ \mathcal{B}_0 \cap \mathcal{C}_{\text{pub}}^{\perp [-1]} &= \mathcal{C}_{\{(m-1, \lambda - 2)\}} \end{aligned}$$

**Line 5,6,7** For  $1 \leq i \leq \lambda - 2$ , we compute

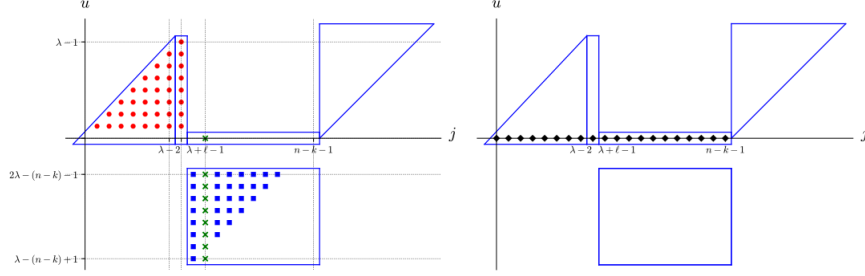
$$\begin{aligned} \mathcal{B}_i &= \mathcal{A} + \sum_{j=0}^{i-1} \mathcal{C}_j^{[i-j]} \\ \mathcal{D}_i &= \mathcal{B}_i \cap \mathcal{C}_{\text{pub}}^{\perp [-1]} = \mathcal{C}_{\{(m-1, \lambda + i - 2)\}} \end{aligned}$$

We prove by induction, suppose that  $\mathcal{D}_i = \mathcal{C}_{\{(m-1, \lambda + i - 2)\}}$  for all  $1 \leq i \leq \ell \leq \lambda - 2$ . We prove this for  $i = \ell + 1$

$$\begin{aligned} \mathcal{B}_{\ell+1} &= \mathcal{A} + \sum_{j=0}^{\ell} \mathcal{C}_j^{[\ell+1-j]} \\ \mathcal{D}_{\ell+1} &= \mathcal{B}_{\ell+1} \cap \mathcal{C}_{\text{pub}}^{\perp [-1]} = \mathcal{C}_{\{(m-1, \lambda + \ell - 1)\}} \end{aligned}$$

Indeed,

$$\begin{aligned} \mathcal{B}_{\ell+1} &= \mathcal{A} + \sum_{j=0}^{\ell} (\mathcal{C}_{\{(m-1, \lambda + j - 2)\}})^{[\ell+1-j]} \\ &= \mathcal{A} + \sum_{j=0}^{\ell} \mathcal{C}_{\{(\ell-j, \lambda + \ell - 1)\}} \\ &= \mathcal{C}_{I_1 \sqcup I_3 - d} + \mathcal{C}_{[0, \ell] \times \{\lambda + \ell - 1\}} \\ &= \mathcal{C}_{I_1 \sqcup (I_3 - d \setminus [\ell - d + 1, \lambda - d - 1] \times \{\lambda + \ell - 1\})} + \mathcal{C}_{* \times \{\lambda + \ell - 1\}} \end{aligned}$$



**Fig. 4.** Points of  $I_1 \sqcup (I_3 - d \setminus [\ell - d + 1, \lambda - d - 1] \times \{\lambda + \ell - 1\}) \sqcup * \times \{\lambda + \ell - 1\}$  and  $\{0\} \times [0, (n - k) - 1]$

where  $*$  is instantiated for the set  $[\ell - d + 1, \lambda - d - 1] \sqcup [0, d]$  of  $\lambda$  distinct integers.

We compute  $\mathcal{D}_{\ell+1}^{[1]} = \mathcal{B}_{\ell+1}^{[1]} \cap \mathcal{C}_{\text{pub}}^\perp$ . Let

$$I_2' = ([0, \lambda - 1] \times \{\lambda - 1\}) \sqcup (\{0\} \sqcup [2 - d, \lambda - d]) \times [\lambda, n - k - 1]$$

and  $M_4 = I_1 \sqcup I_4 \sqcup I_2'$ . Then,  $I_1 + 1, I_3 - d + 1$  and  $* \times \{\lambda + \ell + 1\} \subset M_4$ .

Moreover,

$$\begin{aligned} I_3 - d + 1 \cap \{0\} \times [0, n - k - 1] &= \emptyset \\ I_1 + 1 \cap \{0\} \times [0, n - k - 1] &= \emptyset \end{aligned}$$

Hence, by CodeSet theorem,

$$\begin{aligned} \mathcal{B}_{\ell+1}^{[1]} \cap \mathcal{C}_{\text{pub}}^\perp &= \mathcal{C}_{* \times [\lambda + \ell] \cap \{0\} \times [0, n - k - 1]} \\ &= \mathcal{C}_{\{(0, \lambda + \ell)\}} \end{aligned}$$

Therefore,  $\mathcal{D}_{\ell+1} = \mathcal{B}_{\ell+1} \cap \mathcal{C}_{\text{pub}}^{\perp [-1]} = \mathcal{C}_{\{(m-1, \lambda + \ell - 1)\}}$

**Line 8.** Compute  $\mathcal{H} = \sum_{j=0}^{\lambda-1} \mathcal{C}_j^{[2-j-\lambda]}$ .

We consider the sum of subspace:

$$\begin{aligned} \sum_{j=0}^{\lambda-1} \mathcal{C}_j^{[2-j-\lambda]} &= \sum_{j=0}^{\lambda-2} \mathcal{C}_{\{(m+1-\lambda-j, 0)\}} + \mathcal{C}_{\{(m+1-(n-k), 0)\}} \\ &= \mathcal{C}_{([m+3-2\lambda, m+1-\lambda] \sqcup \{m+1-(n-k)\}) \times \{0\}} = \mathcal{C}_{* \times \{0\}} =: \mathcal{H} \end{aligned}$$

where  $*$  is instantiated for the set  $[m + 2 - 2\lambda, m + 1 - \lambda] \sqcup \{m + 1 - (n - k)\}$  of  $\lambda$  distinct integers.

**Line 9-10** Next, for any  $i \in \{0, \dots, n - k - 1\}$ , one can compute

$$\mathcal{C}_{\text{pub}}^{\perp [-i]} \cap \mathcal{H} = \langle \mathbf{h}_0 + \sum_{i=1}^{\lambda-1} \beta_i^{[-i]} \mathbf{h}_i \rangle$$



- For  $\lambda - 1 \leq i \leq n - k - 1$ ,  $* \times \{i\} \subset M$

$$\{(0, i)\} = (\{0\} \times [0, n - k - 1]) \cap (* \times \{i\})$$

Hence, by the CodeSet theorem,  $\mathcal{C}_{(0,i)} = \mathcal{C}_{\text{pub}}^\perp \cap \mathcal{H}^{[i]}$

- For  $0 \leq i \leq \lambda - 2$ ,  $\mathcal{C}_{\text{pub}}^\perp \text{ }^{[\lambda-1]} = \mathcal{C}_{\{\lambda-1\} \times [\lambda-1, n-k+\lambda-2]}$  and  $\mathcal{H}^{[\lambda+i-1]} = \mathcal{C}_{*\times\{\lambda+i-1\}}$ .  
Moreover,  $* \times \{\lambda + i - 1\} \subset * \times [\lambda - 1, 2\lambda - 3] \subset M$  and

$$\{(\lambda - 1, \lambda + i - 1)\} = (\{\lambda - 1\} \times [\lambda - 1, n - k + \lambda - 2]) \cap (* \times \{\lambda + i - 1\})$$

Hence, by the CodeSet theorem,

$$\begin{aligned} \mathcal{C}_{\text{pub}}^\perp \text{ }^{[\lambda-1]} \cap \mathcal{H}^{[\lambda+i-1]} &= \mathcal{C}_{\{(\lambda-1, \lambda+i-1)\}} \\ \mathcal{C}_{\text{pub}}^\perp \cap \mathcal{H}^{[i]} &= \mathcal{C}_{\{(0,i)\}} \end{aligned}$$

Therefore, for any  $i \in \{0, \dots, n - k - 1\}$ , one can compute

$$\mathcal{C}_{\text{pub}}^\perp \text{ }^{[-i]} \cap \mathcal{H} = \mathcal{C}_{\{(-i,0)\}} = \langle \mathbf{h}_0 + \sum_{i=1}^{\lambda-1} \beta_i^{[-i]} \mathbf{h}_i \rangle$$

□

Note that this specialization of one element of  $\mathcal{S}$  should be true for any element in  $\mathcal{S}$ . Indeed, for 2 elements  $(\mathbf{h}', \beta')$  and  $(\mathbf{h}, \beta)$ , if there exists  $\mathbf{A} \in GL(\lambda, \mathbb{F}_q)$  such that  $(\mathbf{h}', \beta') = \mathbf{A} \cdot (\mathbf{h}, \beta)$ , then  $\langle \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-i]} \mathbf{h}_j \rangle = \langle \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j'^{[-i]} \mathbf{h}'_j \rangle$  since  $\mathbf{h}'_0 + \sum_{j=1}^{\lambda-1} \beta_j'^{[-\ell]} \mathbf{h}'_j = \left( a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} \beta_i \right)^{1-[-\ell]} \left( \mathbf{h}_0 + \sum_{i=1}^{\lambda-1} \beta_i^{[-\ell]} \mathbf{h}_i \right)$

### Second step: Recovering the vector space

From step 1, we recovered the 1-dimensional vector-spaces

$$\forall i = 0, \dots, n - k - 1, \mathcal{A}_i = \left\langle \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-i]} \mathbf{h}_j \right\rangle$$

The vector spaces  $\mathcal{A}_i$  do not depend on  $(\mathbf{h}, \beta) \in \mathcal{S}$ . We introduce the following lemma.

**Lemma 4.** *For any  $\mathbf{u}_0 \in \mathcal{A}_0$ , and for any set  $\mathcal{I} = \{i_1, \dots, i_\lambda\} \subset \{1, \dots, n - k - 1\}$  of  $\lambda$  distinct elements, there exists a unique  $\lambda$ -tuple  $\mathbf{u}_{\mathcal{I}} \stackrel{\text{def}}{=} (\mathbf{u}_{i_1}, \mathbf{u}_{i_2}, \dots, \mathbf{u}_{i_\lambda}) \in \bigotimes_{j=1}^{\lambda} \mathcal{A}_{i_j}$  such that*

$$\sum_{i_j \in \mathcal{I}} \mathbf{u}_{i_j} = \mathbf{u}_0$$

*Proof.* We observe that, from assumption (1) we have

$$\mathcal{A}_{i_1} \oplus \cdots \oplus \mathcal{A}_{i_\lambda} = \langle \mathbf{h}_0, \dots, \mathbf{h}_{\lambda-1} \rangle.$$

Since  $\mathcal{A}_0 \subset \langle \mathbf{h}_0, \dots, \mathbf{h}_{\lambda-1} \rangle$ , this completes the proof.  $\square$

We denote  $k_{i_\ell} \in \mathbb{F}_{q^m}$  such that  $\forall i_\ell \in \mathcal{I}, \mathbf{u}_{i_\ell} = k_{i_\ell} \left( \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-i_\ell]} \mathbf{h}_j \right)$ . A vector  $\mathbf{u}_0 \in \mathcal{A}_0$  can be written under the form

$$\mathbf{u}_0 = \alpha_{\mathbf{h}, \beta} \left( \mathbf{h}_0 + \sum_{j=1}^{\lambda} \beta_j \mathbf{h}_j \right)$$

From the structure of the solution space  $\mathcal{S}$ , there exists an  $(\mathbf{h}, \beta) \in \mathcal{S}$  such that  $\alpha_{\mathbf{h}, \beta} = 1$ . It means that we can fix  $\mathbf{u}_0 := \mathbf{h}_0 + \sum_{j=1}^{\lambda} \beta_j \mathbf{h}_j$  as a known vector. For this element and for any  $\mathcal{I} = \{i_1, \dots, i_\lambda\}$  from Lemma 4 we have

$$\begin{aligned} \sum_{i_\ell \in \mathcal{I}} k_{i_\ell}^{\mathcal{I}} \left( \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-i_\ell]} \mathbf{h}_j \right) &= \left( \sum_{i_\ell \in \mathcal{I}} k_{i_\ell}^{\mathcal{I}} \right) \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \left( \sum_{i_\ell \in \mathcal{I}} k_{i_\ell}^{\mathcal{I}} \beta_j^{[-i_\ell]} \right) \mathbf{h}_j \\ &= \mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j \mathbf{h}_j \end{aligned}$$

Since the  $\mathbf{h}_j$  are linearly independent we obtain the following system

$$(k_{i_1}^{\mathcal{I}}, k_{i_2}^{\mathcal{I}}, \dots, k_{i_\lambda}^{\mathcal{I}}) \begin{bmatrix} 1 & \beta_1^{[-i_1]} & \beta_2^{[-i_1]} & \cdots & \beta_{\lambda-1}^{[-i_1]} \\ 1 & \beta_1^{[-i_2]} & \beta_2^{[-i_2]} & \cdots & \beta_{\lambda-1}^{[-i_2]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \beta_1^{[-i_\lambda]} & \beta_2^{[-i_\lambda]} & \cdots & \beta_{\lambda-1}^{[-i_\lambda]} \end{bmatrix} = (1, \beta_1, \beta_2, \dots, \beta_{\lambda-1})$$

in the unknowns  $k_i^{\mathcal{I}}$  and  $\beta_i$ . From assumption (1), knowing the  $\beta_i$ 's, the solution is unique. To solve the system, let us consider the associate matrix

$$\text{Mat}^{\mathcal{I}}(\vec{X}) := \begin{bmatrix} 1 & X_1^{[i_1]} & X_2^{[i_1]} & \cdots & X_{\lambda-1}^{[i_1]} \\ 1 & X_1^{[i_2]} & X_2^{[i_2]} & \cdots & X_{\lambda-1}^{[i_2]} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & X_1^{[i_\lambda]} & X_2^{[i_\lambda]} & \cdots & X_{\lambda-1}^{[i_\lambda]} \end{bmatrix}$$

where  $\vec{X} = (X_1, X_2, \dots, X_{\lambda-1})$  is formed with the unknowns. We define the multivariate polynomial

$$f^{\mathcal{I}}(\vec{X}) \stackrel{def}{=} \det(\text{Mat}^{\mathcal{I}}(\vec{X}))$$

Since  $f^{\mathcal{I}} \in \mathbb{F}_q[\vec{X}]$  we have

**Lemma 5.**  $f^{\mathcal{I}}$  has degree  $\sum_{j \in \mathcal{I}} [j]$  and for all  $u \in \mathbb{Z}$ ,  $f^{\mathcal{I}+u}(\vec{X}) = f^{\mathcal{I}}(\vec{X})^{[u]}$ .

By Cramer's rule, for any  $j = 1, \dots, \lambda$  we have

$$k_{i_j}^{\mathcal{I}} = \frac{f^{-(\mathcal{I} \setminus \{i_j\}) \cup \{0\}}(\vec{\beta})}{f^{-\mathcal{I}}(\vec{\beta})}, \quad (3)$$

where  $\vec{\beta} = (\beta_1, \dots, \beta_\lambda)$ . Let us define  $\mathcal{J}_s = (\{1, \dots, \lambda + 1\}) \setminus \{s + 1\}$ , for all  $s = 1, \dots, \lambda$ . From (3), we have

$$\forall s \in \{1, \dots, \lambda\}, k_1^{\mathcal{J}_s} = \frac{f^{-(\mathcal{J}_s \setminus \{1\}) \cup \{0\}}(\vec{\beta})}{f^{-\mathcal{J}_s}(\vec{\beta})}$$

By elevating the equation to the power  $[\lambda + 1]$ , from Lemma 5 we have

$$\forall s \in \{1, \dots, \lambda\}, (k_1^{\mathcal{J}_s})^{[\lambda+1]} = \frac{f^{(\lambda+1) - (\mathcal{J}_s \setminus \{1\}) \cup \{0\}}(\vec{\beta})}{f^{(\lambda+1) - \mathcal{J}_s}(\vec{\beta})}$$

Now since we know only the vector space  $\mathcal{A}_1$  and not the exact vectors  $\mathbf{h}_0 + \sum_{j=1}^{\lambda-1} \beta_j^{[-1]} \mathbf{h}_j$ , we do not know  $k_1^{\mathcal{J}_s}$ . However, we can compute the quantity  $k_1^{\mathcal{J}_\lambda} / k_1^{\mathcal{J}_s}$  for  $s \in \{1, \dots, \lambda - 1\}$  thank to Algorithm 2 and Lemma 4.

---

**Algorithm 2:** Determining quotient  $k_1^{\mathcal{J}_\lambda} / k_1^{\mathcal{J}_s}$

---

**Input:**  $\{\mathcal{A}_i\}_{i=1}^{n-k-1}$ ,  $\{\mathcal{J}_s\}_{s=1}^\lambda$  and the vector  $\mathbf{u}_0 \in \mathcal{A}_0$

**Output:**  $\alpha_s = k_1^{\mathcal{J}_\lambda} / k_1^{\mathcal{J}_s}$  for  $s \in \{1, \dots, \lambda - 1\}$

- 1 For  $i = 1, \dots, n - k - 1$ , fix  $\mathbf{u}_i$  arbitrarily in  $\mathcal{A}_i$
  - 2 For  $s = 1, \dots, \lambda$ , find  $a_j^{\mathcal{J}_s}$  such that of  $\sum_{j \in \mathcal{J}_s} a_j^{\mathcal{J}_s} \mathbf{u}_j = \mathbf{u}_0$
  - 3 Return  $\frac{a_1^{\mathcal{J}_\lambda}}{a_1^{\mathcal{J}_s}}$ , for  $s = 1, \dots, \lambda - 1$
- 

Now let us define by  $\alpha_s = (k_1^{\mathcal{J}_\lambda} / k_1^{\mathcal{J}_s})^{[\lambda+1]}$ , for  $s = 1, \dots, \lambda - 1$ . To simplify notations, we also define

$$\forall s \in \{1, \dots, \lambda\} \begin{cases} \mathcal{L}_s = (\lambda + 1) - (\mathcal{J}_s \setminus \{1\}) \cup \{0\} \\ \mathcal{M}_s = (\lambda + 1) - \mathcal{J}_s \end{cases}$$

We obtain the set of equations

$$\forall s \in \{1, \dots, \lambda - 1\}, f^{\mathcal{L}_\lambda}(\vec{\beta}) f^{\mathcal{M}_s}(\vec{\beta}) - \alpha_s f^{\mathcal{M}_\lambda}(\vec{\beta}) f^{\mathcal{L}_s}(\vec{\beta}) = 0$$

Let

$$\mathcal{F}_s(\vec{X}) \stackrel{def}{=} f^{\mathcal{L}_\lambda}(\vec{X}) f^{\mathcal{M}_s}(\vec{X}) - \alpha_s f^{\mathcal{M}_\lambda}(\vec{X}) f^{\mathcal{L}_s}(\vec{X}) \in \mathbb{F}_{q^m}[\vec{X}].$$

The polynomial  $\mathcal{F}_s$  has degree  $q^{\lambda+1} + q^\lambda + 2 \sum_{j=1}^{\lambda-1} q^j + 1 - q^{\lambda-s}$

This gives us a multivariate polynomial system over  $\mathbb{F}_{2^m}$  for which  $\vec{\beta}$  is a solution. However, from our hypotheses we can do better and even reduce the degrees of the polynomials.

Since  $\beta_1, \dots, \beta_\lambda$  are linearly independent they cannot be roots of linear factors over  $\mathbb{F}_q$  of  $\mathcal{F}_s$ . Therefore we can reduce for all  $s$  the polynomial  $\mathcal{F}_s(\vec{X})$  by its  $\mathbb{F}_q$ -linear factors.

**Lemma 6.** *Let us define*

$$f_0(\vec{X}) = \prod_{a \in \mathbb{F}_q} (X_1 + a) \prod_{i=2}^{\lambda-1} \left( \prod_{a_0, \dots, a_{i-1} \in \mathbb{F}_q} (X_i + \sum_{j=1}^{i-1} a_j X_j + a_0) \right)$$

For any set  $\mathcal{I}$  of cardinality  $\lambda$ ,  $f^{\mathcal{I}}(\vec{X})$  is divisible by  $f_0(\vec{X})$

*Proof.* Let  $\vec{\beta}$  be a root of  $X_i + \sum_{j=1}^{i-1} a_j X_j + a_0$  then they are  $\mathbb{F}_q$  co-linear. Hence, for all set of cardinality  $\lambda$ ,  $\mathcal{I}$ , the corresponding columns of  $Mat^{\mathcal{I}}(\vec{\beta})$  are co-linear. Therefore,  $f^{\mathcal{I}}(\vec{\beta}) = \det(Mat^{\mathcal{I}}(\vec{\beta})) = 0$   $\square$

We have the following two corollaries

**Corollary 2.** *We have  $f^{\mathcal{J}_\lambda - 1}(\vec{X}) = f_0(\vec{X})$*

*Proof.* Both polynomials are monic. Since  $\mathcal{J}_\lambda - 1 = \{0, \dots, \lambda - 1\}$ , they also have the same degree  $\sum_{i=0}^{\lambda-1} q^i$   $\square$

**Corollary 3.** *For all  $\mathcal{I} = \{i_1, \dots, i_\lambda\}$ , we have  $(f_0(\vec{X}))^{[i_1]} \mid f^{\mathcal{I}}(\vec{X})$*

We have

- From the lemma:  $f_0(\vec{X})$  divides  $f^{\mathcal{M}_s}(\vec{X})$  and  $f^{\mathcal{L}_s}(\vec{X})$  for all  $s \in \{1, \dots, \lambda - 1\}$ .
- From corollary 3 :  $f_0(\vec{X})^{[1]}$  divides  $f^{\mathcal{M}_\lambda}(\vec{X})$  and  $f^{\mathcal{L}_\lambda}(\vec{X})$ , since the minimum index of the sets is equal to 1.

Therefore, for all  $s \in \{1, \dots, \lambda - 1\}$ ,  $\mathcal{F}_s(\vec{X})$  can be divided by  $f_0(\vec{X})^{q+1}$ . We now consider the reduced polynomials

$$\forall s \in \{1, \dots, \lambda - 1\}, \quad \mathcal{P}_s(\vec{X}) \stackrel{def}{=} \frac{\mathcal{F}_s(\vec{X})}{(f_0(\vec{X}))^{q+1}} = \frac{\mathcal{F}_s(\vec{X})}{f^{\mathcal{J}_\lambda - 1}(\vec{X}) f^{\mathcal{J}_\lambda}(\vec{X})}$$

This gives us a new polynomial system for which  $\vec{\beta}$  is also a solution, but the degree is reduced.

**Lemma 7.** Let  $\vec{A} = (a_{i,j})_{i=0,j=0}^{\lambda-1,\lambda-1} \in \mathbf{PGL}(\lambda; \mathbb{F}_q)$ . Consider the transformation on  $f^{\mathcal{I}}(\vec{X})$  defined on  $\vec{X} = (X_1, \dots, X_{\lambda-1})$  by

$$\forall j \in \{1, \dots, \lambda-1\}, \quad X_j \mapsto \frac{a_{0,j} + \sum_{i=1}^{\lambda-1} a_{i,j} X_i}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} X_i}$$

then the polynomial  $f^{\mathcal{I}}(\vec{X})$  is transformed into

$$f^{\mathcal{I}}(\vec{X}) \mapsto \vec{A} \cdot f^{\mathcal{I}}(\vec{X}) \stackrel{\text{def}}{=} \frac{\Delta_A}{(a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} X_i)^{\deg(f^{\mathcal{I}})}} f^{\mathcal{I}}(\vec{X})$$

where  $\Delta_A$  is the determinant of  $\mathbf{A}$ .

*Proof.* Let  $D = a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} X_i$ . Thus, for  $j = 1, \dots, \lambda$ , the  $j$ th row of  $\text{Mat}^{\mathcal{I}}(\vec{X})$  denoted by  $\text{Row}_j(\text{Mat}^{\mathcal{I}}(\vec{X}))$  becomes

$$\text{Row}_j(\text{Mat}^{\mathcal{I}}(\vec{X})) \mapsto \frac{\text{Row}_j(\text{Mat}^{\mathcal{I}}(\vec{X}) \cdot \mathbf{A})}{D^{[i_j]}}.$$

Therefore, since  $\deg(f^{\mathcal{I}}) = \sum_{j \in \mathcal{I}} [j]$ , from lemma 5, we obtain

$$\begin{aligned} \det(\text{Mat}^{\mathcal{I}}(\vec{X})) &\mapsto \frac{\det \vec{A}}{D^{\deg(f^{\mathcal{I}})}} \det(\text{Mat}^{\mathcal{I}}(\vec{X})) \\ f^{\mathcal{I}}(\vec{X}) &\mapsto \frac{\Delta_A}{D^{\deg(f^{\mathcal{I}})}} f^{\mathcal{I}}(\vec{X}) \end{aligned}$$

□

Apply the lemma, we have

$$\begin{aligned} \mathcal{F}_s(\vec{X}) &\mapsto \frac{\Delta_A^2}{D^{q^{\lambda+1} + q^\lambda + 2 \sum_{j=1}^{\lambda-1} q^{j+1} - q^{\lambda-s}}} \mathcal{F}_s(\vec{X}) \\ f^{\mathcal{J}_{\lambda-1}}(\vec{X}) &\mapsto \frac{\Delta_A}{D^{\sum_{j=0}^{\lambda-1} q^j}} f^{\mathcal{J}_{\lambda-1}}(\vec{X}) \\ f^{\mathcal{J}_\lambda}(\vec{X}) &\mapsto \frac{\Delta_A}{D^{\sum_{j=1}^{\lambda} q^j}} f^{\mathcal{J}_\lambda}(\vec{X}) \end{aligned}$$

Hence,

$$\mathcal{P}_s(\vec{X}) \mapsto \frac{1}{D^{q^{\lambda+1}-q^{\lambda-s}}} \mathcal{P}_s(\vec{X})$$

We therefore have

**Proposition 3.** *If there isn't any common factor between the polynomials  $\mathcal{P}_s(X)$ , then the set of root of the polynomial system*

$$\forall i = 1, \dots, \lambda - 1, \quad \mathcal{P}_i(\vec{X}) = 0 \quad (4)$$

*equals the orbit of any root under the group action of  $\mathbf{PGL}(\lambda, \mathbb{F}_q)$*

*Proof.* If there isn't any common factor between the polynomials  $\mathcal{P}_s(X)$  then the number of roots is at bounded by  $\prod_{j=1}^{\lambda-1} (q^{\lambda+1} - q^j) = |\mathbf{PGL}(\lambda, q)|$  (Bezout bound [8]). Moreover, any element in the orbit of a solution  $\beta$  under the group action of  $\mathbf{PGL}(\lambda, \mathbb{F}_q)$  is again root of the system. From Assumption (3) the orbit of  $\beta$  under  $\mathbf{PGL}(\lambda, \mathbb{F}_q)$  has cardinality  $= |\mathbf{PGL}(\lambda, \mathbb{F}_q)|$  which means that the stabilization of  $\beta$  with respect to this group action is trivial. In that case any root of the system (4) corresponds to an element of  $\mathcal{S}$ .  $\square$

For instance, when  $q = 2$  and  $\lambda = 3$  the system of equation below taking  $(\beta_1, \beta_2)$  as solution:

$$\begin{cases} Pr_1(X, Y) = 0 \\ Pr_2(X, Y) = 0 \end{cases}$$

This is a system of 2 polynomial equation in 2 variables. In practice, by using MAGMA, we can see that there isn't any common factor between  $Pr_1(X)$  and  $Pr_2(X)$ . Therefore, the number of roots has Bezout's upper bound by the product of the degrees of  $Pr_1(X, Y)$  and  $Pr_2(X, Y)$ .

Therefore, the number of roots are at most  $(q^4 - q)(q^4 - q^2) = |\mathbf{PGL}(3, q)|$ . Thus, all the roots are in the orbit of a root under an action of  $\mathbf{PGL}(3, q)$

The remaining problem is finding a root of the system of equation above. It can be done by the following steps:

1. Calculating  $Res(Pr_1, Pr_2, Y)$  the resultant of  $Pr_1$  and  $Pr_2$  in the variable  $Y$ . We obtain a univariate polynomial of degree 168 in variable  $X$ . Finding one root  $x_0$  of this polynomial.
2. Calculating  $\gcd(Pr_1(x_0, Y), Pr_2(x_0, Y))$  which is a polynomial of degree 4 in variable  $Y$ . Taking one root  $y_0$  and verify it is a root of the system of equation.

In general, the problem of finding one root of a system of polynomial equation is a hard question as well as finding all roots of a system of polynomial equation.

**Polynomial System Solving over Finite Fields** Let  $\mathbb{F}$  is a finite field.  
**Input:**  $f_1(x_1, \dots, x_n), \dots, f_m(x_1, \dots, x_n) \in \mathbb{F}[x_1, \dots, x_n]$ .  
**Goal:** Find a vector  $\alpha = (\alpha_1, \dots, \alpha_n) \in \mathbb{F}^n$  s.t:  $f_1(\alpha) = \dots = f_m(\alpha) = 0$ .

Theoretically it is a NP-hard problem (problem AN9 p.251 in Appendix: A list of NP-complete problem [9]). For the special cases  $\lambda = 2$  and  $\lambda = 3$  finding a solution can be done in polynomial-time (by using properties of resultants for  $\lambda = 3$ ).

However, in the case of no common factor, the number of roots is bounded by Bezout bound. To check that  $Pr_i(X)_{i=1}^{\lambda-1}$  don't have common factor, we can check whether  $Res(Pr_i(X), Pr_j(X), X_1) \neq 0, \forall 1 \leq i < j \leq \lambda - 1$ . (Prop. 1, Ch. 3, [10]). It costs  $O(d^3)$  where  $d = \prod_{j=1}^{\lambda-1} (q^{\lambda+1} - q^j)$  arithmetic operations over  $\mathbb{F}_{q^m}[X_2, \dots, X_{\lambda-1}]$ .

We can see the importance of the Assumption(3) in the Proposition 3. In the case where this assumption does not satisfy, i.e there exists  $\overline{\mathbf{A}} \in \mathbf{PGL}(\lambda, \mathbb{F}_q) \setminus \overline{\mathcal{I}}_\lambda$  and  $\mathbf{A} = (a_{ij})_{i,j=1}^\lambda$  such that

$$\beta_j = \frac{a_{0,j} + \sum_{i=1}^{\lambda-1} a_{i,j} \beta_i}{a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} \beta_i}$$

Thus,  $\beta$  is a root of a system of  $\lambda - 1$  polynomial equations of degree 2:

$$(a_{0,0} + \sum_{i=1}^{\lambda-1} a_{i,0} X_i) X_j - (a_{0,j} + \sum_{i=1}^{\lambda-1} a_{i,j} X_i) = 0, \quad j = 1, \dots, \lambda - 1$$

This polynomial is different from 0. Indeed, if it was,  $a_{0,0} = a_{j,j}$  for  $j = 1, \dots, \lambda - 1$  and  $a_{i,j} = 0$  for  $i \neq j$ , which means  $\mathbf{A} \in \overline{\mathcal{I}}_\lambda$ .

This system is multivariate quadratic (MQ)-system, the associated problem to decide if this system is solvable or not, also known as MQ-problem, is proven to be NP-complete [9]. Some algorithms used to solve this system is reviewed in the paper [11]. In case of  $\lambda = 3$ , this can be solved easily by Resultant. Therefore, when the Assumption (3) does not satisfy, we can exploit some information about  $\beta$  by solving a multivariate quadratic system.

**Final step:** Now from a solution  $\vec{\beta}$  to (4), we aim at finding the corresponding vector  $\vec{h} \in (\mathbb{F}_{q^m}^n)^\lambda$  such that  $(\vec{h}, \vec{\beta}) \in \mathcal{S}$ .

We point out the key steps in the Coggia-Couvreur attack for  $\lambda$  as follows. To be convenient, we denote known elements by blue color and unknown elements by red color. Given  $\beta'_1, \dots, \beta'_{\lambda-1}$ , recover  $(h'_0, \dots, h'_{\lambda-1}, \beta'_1, \dots, \beta'_{\lambda-1})$  corresponding.

1. For  $\mathcal{I} = \{1, \dots, \lambda\}$ , since  $\vec{\beta}^i$  is known,  $k_i = \frac{f^{-(\mathcal{I} \setminus \{i\}) \cup \{0\}}(\vec{\beta}^i)}{f^{-\mathcal{I}}(\vec{\beta}^i)}$ ,  $i = 1, \dots, \lambda$  can be computed. Moreover, from the Lemma 4, there exists a unique  $\lambda$ -tuple  $\mathbf{u}_{\mathcal{I}} = (\mathbf{u}_1, \dots, \mathbf{u}_\lambda) \in \prod_{j=1}^{\lambda} \mathcal{A}_j$  such that  $\sum_{i=1}^{\lambda} \mathbf{u}_i = \mathbf{u}_0$ , so we can compute

$$\mathbf{h}'_0 + \sum_{j=1}^{\lambda-1} \beta_j'^{[-i]} \mathbf{h}'_j = \frac{\mathbf{u}_i}{k_i}, i = 1, \dots, \lambda.$$

$$(\mathbf{h}'_0, \dots, \mathbf{h}'_{\lambda-1}) \begin{bmatrix} 1 & 1 & \dots & 1 \\ \beta_1'^{[-1]} & \beta_1'^{[-2]} & \dots & \beta_1'^{[-\lambda]} \\ \vdots & \vdots & \ddots & \vdots \\ \beta_{\lambda-1}'^{[-1]} & \beta_{\lambda-1}'^{[-2]} & \dots & \beta_{\lambda-1}'^{[-\lambda]} \end{bmatrix} = \left( \frac{\mathbf{u}_1}{k_1}, \frac{\mathbf{u}_2}{k_2}, \dots, \frac{\mathbf{u}_\lambda}{k_\lambda} \right)$$

It implies to a linear system of  $\lambda$  equation and  $\lambda$  unknowns which are vectors  $\mathbf{h}'_0, \dots, \mathbf{h}'_{\lambda-1}$  and the determinant of the matrix of coefficients is non-zero.

2. After recovering an alternate key of the form  $(\mathbf{h}'_0, \dots, \mathbf{h}'_{\lambda-1}, \beta_1', \dots, \beta_{\lambda-1}')$ , we can compute the dual code  $\mathcal{C}_{\text{pub}}^\perp$  and hence decrypt the ciphertext.

### 2.3 Complexity of the case $\lambda = 3$

This part shows the complexity of the attack by giving the number of operation in  $\mathbb{F}_{q^m}$ . Let  $\omega$  be the exponent of the complexity of linear algebra operations. The Frobenius map costs  $O(\log q)$  operations.

#### Step 1.

- Computation of dual code  $\mathcal{C}_{\text{pub}}^\perp$  costs  $O(n^\omega)$  operations.
- Computation of  $\mathcal{C}_{\text{pub}}^{\perp [i]}$ ,  $\forall i = 1, \dots, n - k + 1$  costs  $O(n^2 \log q)$  operations.
- Computation  $S_j = \sum_{i=j}^{j+\lambda-1} \mathcal{C}_{\text{pub}}^{\perp [i]}$  uses Gaussian elimination, so it costs  $O(n^\omega)$ .

Thus, computation  $\bigcap_{i=0}^{n-k-\lambda+1} S_j$  costs  $O(n^{\omega+1})$ .

Overall step 1 costs  $O(n^3 \log q + n^{\omega+1})$  operations.

#### Step 2.

- Computation  $(u_1^{\mathcal{I}}, \dots, u_\lambda^{\mathcal{I}})$  represents the resolution of a linear system  $\lambda$  unknowns and  $n$  equations costs  $O(n)$  operations. This computation performed  $O(n)$  times, so it costs  $O(n^2)$  operations.
- Complexity of finding a root of a polynomial of degree  $\tilde{d}$  by Cantor–Zassenhaus algorithm ([12]) costs  $\tilde{O}(\tilde{d}^2 m \log q)$  operations in  $\mathbb{F}_{q^m}$  for  $\tilde{d} = (q^4 - q)(q^4 - q^2)$ .
- Computation of resultant of bivariate polynomials  $\text{Res}(P_1, P_2, X)$  which  $P_1, P_2$  of degree  $d, e$  by Lickteig–Roy subresultant algorithm costs  $O(d^2 e)$  ([13]).
- A finite number of linear systems solving costs  $O(n^\omega)$

**Summary.** For  $m = O(n)$ , overall cost of  $O(n^3 \log q + n^{\omega+1}) + \tilde{O}(d^2 n \log q)$  for  $d = (q^4 - q)(q^4 - q^2)$ .



**Conclusion** We provided a distinguisher for the Loidreau’s scheme for any  $\lambda$  and the public code has rate  $R_{pub} \geq 1 - 1/\lambda$ . From this distinguisher, we are able to complete a polynomial time key recovery with the assumption of finding one root of a system of polynomial equation. Moreover, we have extended the key recovery attack for  $\lambda = 3$ .

The parameters of  $(k, n)$ , which  $R_{pub} \geq 1 - 1/\lambda$  should be avoided in Loidreau’s scheme. In the future, it will be worthwhile to attempt a modification of the attack to work for lower rate codes  $R_{pub} < 1 - 1/\lambda$  as well.

## References

1. R. J. McEliece, “A Public-Key Cryptosystem Based On Algebraic Coding Theory,” *Deep Space Network Progress Report*, vol. 44, pp. 114–116, Jan. 1978.
2. E. M. Gabidulin, A. V. Paramonov, and O. V. Tretjakov, “Ideals over a non-commutative ring and their application in cryptology,” in *Advances in Cryptology — EUROCRYPT ’91* (D. W. Davies, ed.), (Berlin, Heidelberg), pp. 482–489, Springer Berlin Heidelberg, 1991.
3. P. Gaborit, G. Murat, O. Ruatta, and G. Zemor, “Low Rank Parity Check codes and their application to cryptography,” in *The International Workshop on Coding and Cryptography (WCC 13)* (L. Budaghyan, T. Helleseth, and M. G. Parker, eds.), (Bergen, Norway), p. 13 p., Apr 2013. ISBN 978-82-308-2269-2.
4. R. Overbeck, “Structural attacks for public key cryptosystems based on gabidulin codes,” *J. Cryptology*, vol. 21, pp. 280–301, 2008.
5. P. Loidreau, “A new rank metric codes based encryption scheme,” in *PQCrypto 2017* (T. Lange and T. Takagi, eds.), vol. 10346 of *Lecture Notes in Computer Science*, (Utrecht, Netherlands), pp. 3–17, Springer, June 2017.
6. D. Coggia and A. Couvreur, “On the security of a loidreau’s rank metric code based encryption scheme,” *CoRR*, vol. abs/1903.02933, 2019.
7. A. Ghatak, “Extending coggia-couvreur attack on loidreau’s rank-metric cryptosystem,” 2020.
8. J.-C. Faugère, P. Gaudry, L. Huot, and G. Renault, “Polynomial Systems Solving by Fast Linear Algebra.” 27 pages, Apr. 2013.
9. M. R. Garey and D. S. Johnson, *Computers and Intractability; A Guide to the Theory of NP-Completeness*. USA: W. H. Freeman Co., 1990.
10. D. A. Cox, J. Little, and D. O’Shea, *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3/e (Undergraduate Texts in Mathematics)*. Berlin, Heidelberg: Springer-Verlag, 2007.
11. E. Thomae and C. Wolf, “Solving systems of multivariate quadratic equations over finite fields or: From relinearization to mutantxl,” 2010.
12. A. Bostan, F. Chyzak, M. Giusti, R. Lebreton, G. Lecerf, B. Salvy, and É. Schost, *Algorithmes Efficaces en Calcul Formel*. Palaiseau: Frédéric Chyzak (auto-édit.), Sept. 2017. 686 pages. Imprimé par CreateSpace. Aussi disponible en version électronique.
13. G. Lecerf, “On the complexity of the Lickteig-Roy subresultant algorithm.” working paper or preprint, Jan. 2017.