Algebraic Attack against Variants of McEliece with Goppa Polynomial of a Special Form

Jean-Charles Faugère, Ludovic Perret, Frédéric De Portzamparc

Algebraic Attack against Variants of McEliece with Goppa Polynomial of a Special Form

Jean-Charles Faugère[1,2,3], Ludovic Perret[2,1,3], and
Frédéric de Portzamparc[4,1,2,3]

INRIA, Paris-Rocquencourt Center[1],
UPMC Univ Paris 06, UMR 7606, LIP6, F-75005, Paris, France[2]
CNRS, UMR 7606, LIP6, F-75005, Paris, France[3]
Gemalto, 6 rue de la Verrerie 92190, Meudon, France[4]
jean-charles.faugere@inria.fr, ludovic.perret@lip6.fr,
frederic.urvoydeportzamparc@gemalto.com

Abstract. In this paper, we present a new algebraic attack against some special cases of Wild McEliece Incognito, a generalization of the original McEliece cryptosystem. This attack does not threaten the original McEliece cryptosystem. We prove that recovering the secret key for such schemes is equivalent to solving a system of polynomial equations whose solutions have the structure of a usual *vector space*. Consequently, to recover a basis of this vector space, we can greatly reduce the number of variables in the corresponding algebraic system. From these solutions, we can then deduce the basis of a GRS code. Finally, the last step of the cryptanalysis of those schemes corresponds to attacking a McEliece scheme instantiated with particular GRS codes (with a polynomial relation between the support and the multipliers) which can be done in polynomial-time thanks to a variant of the Sidelnikov-Shestakov attack. For Wild McEliece & Incognito, we also show that solving the corresponding algebraic system is notably easier in the case of a non-prime base field $\mathbb{F}_q$. To support our theoretical results, we have been able to practically break several parameters defined over a non-prime base field $q \in \{9, 16, 25, 27, 32\}$, $t \leqslant 6$, extension degrees $m \in \{2, 3\}$, security level up to 2^{129} against information set decoding in few minutes or hours.

Keywords. public-key cryptography, McEliece cryptosystem, algebraic cryptanalysis.

1 Introduction

Algebraic cryptanalysis is a general attack technique which reduces the security of a cryptographic primitive to the difficulty of solving a non-linear system of equations. Although the efficiency of general polynomial system solvers such as Gröbner bases, SAT solvers ..., is constantly progressing such algorithms all face the intrinsic hardness of solving polynomial equations. As a consequence, the success of an algebraic attack relies crucially in the ability to find the best modelling in term of algebraic equations.

In [14,15], Faugère, Otmani, Perret and Tillich (FOPT) show – in particular – that the key-recovery of McEliece [20] can be reduced to the solving of a system of non-linear equations. This key-recovery system can be greatly simplified for so-called compact variants of McEliece, e.g. [4,21,2,16,23,1], leading to an efficient attack against various compact schemes [14,13]. However, it is not clear whether the attack of [14,15] could be efficient against non-compact variants of McEliece, the bottleneck being the huge number of variables and the high degree of the equations involved in the algebraic modelling.

We present a novel algebraic modelling that applies to the original McEliece system and to generalizations such as *Wild McEliece* [6] and *Wild McEliece Incognito* [8]. Note, however, that the resulting attack works only in some special cases, and in particular does not work for the original McEliece system. Wild McEliece uses *Wild Goppa codes*, that is Goppa codes over $\mathbb{F}_q, q \geqslant 2$, with a Goppa polynomial of the form Γ^{q-1} (Γ being an univariate polynomial of low degree). This form of the Goppa polynomial, generalizing the form used in the original McEliece system for $q = 2$, allows to increase the number of errors that can be added to a message (in comparison to a random Goppa polynomial of the same degree). In [8], Bernstein, Lange, and Peters generalized this idea by using Goppa polynomials of the form $f\,\Gamma^{q-1}$, with f another univariate polynomial. We shall call such Goppa codes *Masked Wild Goppa* codes. Like the authors of [8], we refer to this version as Wild McEliece *Incognito*. All in all, Wild McEliece/Wild McEliece Incognito allow the users to select parameters with a resistance to all known attacks, so in particular to the algebraic attack of [14,15], similar to that of binary Goppa codes but with much smaller keys. The security of Wild McEliece defined over quadratic extension has been recently investigated in [11], where the authors presented a polynomial time attack on the key when $t = \deg(\Gamma) > 1$.

1.1 Our Contributions

We present a completely new algebraic attack dedicated to Wild McEliece and Wild McEliece Incognito. To do so, we show that the key-recovery for such schemes is equivalent to finding the basis of a vector-space which is hidden in the zero-set of an algebraic system. To our knowledge, this is a new computational problem that never appeared in algebraic cryptanalysis before. Compared to the algebraic attack proposed in [14] for McEliece, our modelling intrinsically involves less variables. Informally, the multiplicity of the Goppa polynomial implies that the solutions of the algebraic system considered here have a structure of vector space. When the base field is $\mathbb{F}_q$ with $q > 2$, this simplifies its resolution. For instance, for a Wild McEliece Incognito scheme with parameters $q = 32, m = 2, n = 864, t = 2, deg(f) = 36$), we end up with an algebraic system having only 9 variables ([14] would require to consider algebraic equations with 1060 variables in the same situation). On a very high level, our attack proceeds in two main steps.

1. **Polynomial system solving.** We have to solve a non-linear system of equations whose zero-set forms, unexpectedly, a vector space of some known

dimension d. Consequently, we can reduce the number of variables by fixing d variables in the initial and repeat several times the solving step to recover a basis of the vector space solution. This is the most computationally difficult part of the attack.

2. **Linear algebra to recover the secret key.** The second phase is the treatment of the solutions obtained at the first step so as to obtain a private description which allows to decode the public-key as efficiently as the private key. It involves computing intersections of vector spaces, solving linear systems, and polynomial interpolation. Thus, this part can be done efficiently, i.e. in polynomial time.

We detail below the main ingredients of our attack.

An Algebraic Modelling with a Vector Space Structure on the Zero Set. Let $\mathbf{G}_{pub} = (g_{i,j})_{\substack{0 \leqslant i \leqslant n-1 \\ 0 \leqslant j \leqslant k-1}} \in \mathbb{F}_q^{k \times n}$ be the public matrix of a Wild McEliece Incognito scheme. We denote by m its extension degree, and set $t = \deg(\Gamma)$. Our attack considers the system

$$\mathcal{W}_{q,a}(\mathbf{Z}) = \bigcup_{u \in \mathcal{P}_a} \left\{ \sum_{j=0}^{n-1} g_{i,j} Z_j^u = 0 \quad | \quad 0 \leqslant i \leqslant k-1 \right\}, \tag{1}$$

with $\mathcal{P}_a = \{1, 2, \ldots, p^a - 1\} \cup \{p^a, p^{a+1}, \ldots, q\}$ being a subset of $\{1, \ldots, q\}$.

As a comparison, the modelling of Faugère, Otmani, Perret and Tillich [14] will necessarily introduce variables $\mathbf{X} = (X_0, \ldots, X_{n-1})$, $\mathbf{Y} = (Y_0, \ldots, Y_{n-1})$ and $\mathbf{W} = (W_0, \ldots, W_{n-1})$ for all the support and multipliers (that is, the vectors $\mathbf{y} = \Gamma(\mathbf{x})^{-1}$ and $\mathbf{w} = f(\mathbf{x})^{-1}$). In [14], the system is as follows:

$$\bigcup_{0 \leqslant u \leqslant t-1} \left\{ \sum_{j=0}^{n-1} g_{i,j} Y_j X_j^u = 0 \quad | \quad 0 \leqslant i \leqslant k-1 \right\}.$$

In our context, [14] would induce a system containing monomials of the forms $Y_i^{\ell_Y} X_i^{\ell_X}$ and even $Y_i^{\ell_Y} X_i^{\ell_X} W_i^{\ell_W}$ (for some ℓ_X, ℓ_Y, ℓ_W). Here, we use a single vector of variables $\mathbf{Z} = (Z_0, \ldots, Z_{n-1})$ and write very simple homogeneous equations. The secret-key $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{w}$ will be recovered from $\mathbf{Z}$, but in a second step. The main advantage of this approach (Theorem 2) is that the solutions of $\mathcal{W}_{q,a}(\mathbf{Z})$ have a very unexpected property for a non-linear system: they form a vector space. This allows to reduce the number of "free" unknowns in $\mathcal{W}_{q,a}(\mathbf{Z})$ by the dimension of the solutions. For example, we end up with a system containing only 9 variables for an Incognito scheme with parameters $q = 32, m = 2, n = 864, t = 2, deg(f) = 36$). The algebraic description of Goppa codes proposed in [14] would require to consider algebraic equations with 1060 variables for the same parameters.

To be more precise, the vector space underlying the solutions of (1) is closely related to Generalized Reed-Solomon (GRS) codes.

Definition 1 (Generalized Reed-Solomon codes). *Let* $\mathbf{x} = (x_0, \ldots, x_{n-1}) \in (\mathbb{F}_{q^m})^n$ *where all* x_i*'s are distinct and* $\mathbf{y} = (y_0, \ldots, y_{n-1}) \in \left(\mathbb{F}_{q^m}^*\right)^n$*. The* Generalized Reed-Solomon code *of dimension* t*, denoted by* $\mathrm{GRS}_t(\mathbf{x}, \mathbf{y})$*, is defined as follows*

$$\mathrm{GRS}_t(\mathbf{x}, \mathbf{y}) \stackrel{def}{=} \left\{(y_0 Q(x_0), \ldots, y_{n-1} Q(x_{n-1})) \mid Q \in \mathbb{F}_{q^m}[z], \deg(Q) \leqslant t-1\right\}.$$

We shall call $\mathbf{x}$ *the* support *of the code, and* $\mathbf{y}$ *the* multipliers*.*

Theorem 2 shows that the solutions of $\mathcal{W}_{q,a}(\mathbf{Z})$ contain a vector-space which is generated by sums of codewords of Generalized Reed-Solomon (GRS) codes $\mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)$ (where $(\mathbf{x}, \mathbf{y})$ is a key equivalent to the secret key). In Section 3.2, we explain more precisely how we can take advantage of this special structure for solving (1) and recover a basis of the vector subspace.

A Method to Isolate a GRS **Code From a Sum of** GRS**.** From a basis of this sum of GRS, we want to recover the basis of the code $\mathrm{GRS}_t(\mathbf{x}, \mathbf{y})$. We refer to this phase as the *disentanglement*. We expose our solution in Section 4, which relies on a well-chosen intersection of codes. It is rigorously proved in characteristic 2 (Proposition 6). For other characteristics, we launched more than $100,000$ experiments and observed that Proposition 6 still held in all cases.

A Sidelnikov-Shestakov-Like Algorithm Recovering the Goppa Polynomial. Given a basis of a Generalized Reed-Solomon code $\mathrm{GRS}_t(\mathbf{x}, \mathbf{y})$, the Sidelnikov-Shestakov attack [26] consists in recovering the secret pair of vectors $(\mathbf{x}, \mathbf{y})$. It is well-known that the Sidelnikov-Shestakov attack works in polynomial-time. In our case, we have to address a slight variant of this problem. There is a polynomial relation $\Gamma(z)$ linking $\mathbf{x}$ and $\mathbf{y}$ which is part of the private key. In Section 4.2, we provide an adaptation of [26] to obtain a key $(\mathbf{x}', \mathbf{y}', \Gamma')$ equivalent to the secret key, also in polynomial time. We are unaware of such an algorithm published so far.

A Weakness of Codes Defined Over non-prime Base Fields. Independently of our algebraic attack, we prove a general result about Goppa codes defined over $\mathbb{F}_q$ (with $q = p^s$, p prime and $s > 0$) and whose polynomials have a factor $\Gamma(z)$ with multiplicity q. We show in Section 5 that the coordinate-vectors over $\mathbb{F}_p$ of the codewords of such a public code are codewords of a Wild Goppa code, defined over $\mathbb{F}_p$, with same secret support and Goppa polynomial $\Gamma(z)^p$ (Theorem 8). In other words, this construction gives access, from the public key, to a new code implying the same private elements. As a consequence, using non-prime base fields reveals more information on the secret key than expected by the designers. Any key-recovery attack can benefit from it. This is then an intrinsic weakness of Goppa codes defined over non-prime base fields. In our context, this property provides additional linear equations between the variables Z_j's of the system (1). We can reduce the number of variables from $(p^s - 1)mt$ to $(p-1)mst$ essential variables, and make the codes defined over fields $\mathbb{F}_q$ with $q = p^s$ notably weaker (Corollary 10).

1.2 Impact of our Work

In order to evaluate the efficiency of our attack, we considered various parameters for which [6] said that strength is "unclear" and that an attack would not be a "surprise" but for which no actual attack was known.

Information Set Decoding (ISD) is a generic decoding technique which allows message-recovery. This technique has been intensively studied since 1988 (e.g. [17,10,5,7,19,3]) and remains the reference to choose secure parameters in code-based cryptography. The latest results from [24] have been used to generate the parameters for Wild McEliece and Wild McEliece Incognito.

In [6, Table 7.1] numerous keys are presented which illustrate the key size reduction when the size of the field q grows. Another consequence of increasing q is pointed out by the authors of [6]: the low number of irreducible polynomials in $\mathbb{F}_{q^m}[z]$ entails a possible vulnerability against the SSA structural attack ([18,25]). Although the designers provide a protection (using non full-support codes) such that [18] is completely infeasible today, they warn that further progress in [18] may jeopardize the parameters with $q > 9$ and thus estimate that those parameters have unclear security. Our experiments reveal that, in the case of non-prime base fields, it is already possible to recover the secret key in some minutes with our attack using off-the-shelf tools (MAGMA [9] V2.19-1).

Getting around the alleged vulnerability against SSA was the main motivation for proposing Incognito: in [8, Table 5.1], they propose parameters considered fully secure, as all ISD-complexities are above 2^{128} and numbers of possible Goppa polynomials greater than 2^{256}. It turns out that, in the case of non-prime base fields, the extra-shield introduced in Incognito is not a protection against our attack. We can practically break the recommended parameters for $q \in \{16, 27, 32\}$. However, we could not solve (in less than two days) the algebraic systems involved for extension degrees $m \geqslant 4$ or $t \geqslant 7$, and for codes over $\mathbb{F}_p$, p prime. So, it does not threaten the original McEliece cryptosystem. To conclude, we highlight that Theorem 2 is valid for all Goppa codes whose Goppa polynomial has multiplicities and should be then taken into account by designers in the future. Figure 1 provides a diagram which recapitulates all the steps performed to solve the system (1) and recover the secret key.

2 Coding Theory Background

Let $\mathbb{F}_q$ be a finite field of $q = p^s$ elements (p prime, and $s > 0$). To define conveniently the various kinds of codes we will deal with, we introduce the following Vandermonde-like matrices:

$$\mathbf{V}_t(\mathbf{x}, \mathbf{y}) \stackrel{\text{def}}{=} \begin{pmatrix} y_0 & \cdots & y_{n-1} \\ y_0 x_0 & \cdots & y_{n-1} x_{n-1} \\ \vdots & & \vdots \\ y_0 x_0^{t-1} & \cdots & y_{n-1} x_{n-1}^{t-1} \end{pmatrix}, \tag{2}$$

where $\big(\mathbf{x} = (x_0, \ldots, x_{n-1}), \mathbf{y} = (y_0, \ldots, y_{n-1})\big) \in \mathbb{F}_{q^m}^n \times \mathbb{F}_{q^m}^n$.

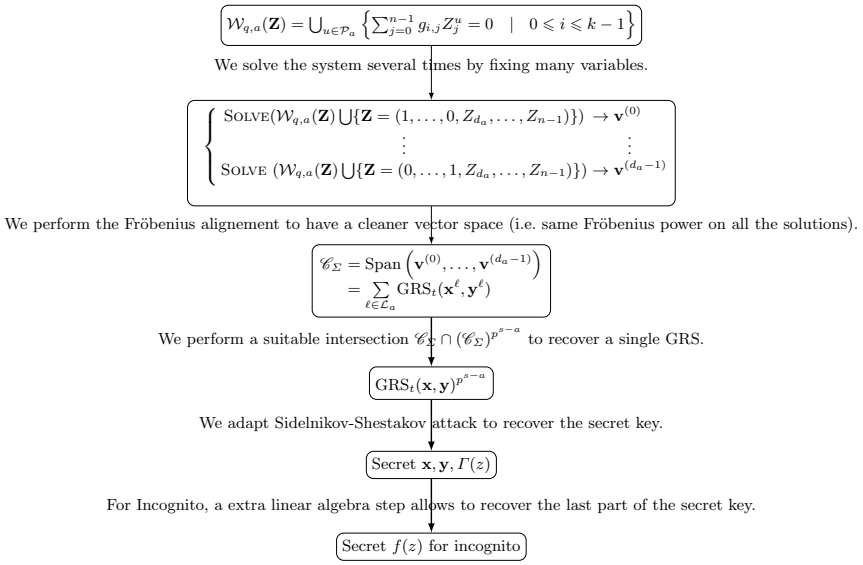

Fig. 1. Overview of the attack.

With suitable $\mathbf{x}$ and $\mathbf{y}$, the rows of such matrices $\mathbf{V}_t(\mathbf{x},\mathbf{y})$ define Generalized Reed-Solomon (GRS) codes (Definition 1). Alternant and Goppa codes can be viewed as the restriction of duals of GRS codes to the base field $\mathbb{F}_q$.

Definition 2 (Alternant/Goppa Codes). *Let* $\mathbf{x} = (x_0,\ldots,x_{n-1}) \in (\mathbb{F}_{q^m})^n$ *where all* x_i*'s are distinct and* $\mathbf{y} \in (\mathbb{F}_{q^m}^*)^n$*. The* alternant code *of order* t *is defined as* $\mathcal{A}_t(\mathbf{x},\mathbf{y}) \stackrel{def}{=} \left\{\mathbf{c}\in\mathbb{F}_q^n \mid \mathbf{V}_t(\mathbf{x},\mathbf{y})\mathbf{c}^T = \mathbf{0}\right\}$ *. As for* GRS *codes,* $\mathbf{x}$ *is the* support*, and* $\mathbf{y}$ *the* multipliers*. Let* $g(z) \in \mathbb{F}_{q^m}[z]$ *be of degree* t *satisfying* $g(x_i) \neq 0$ *for all* $i, 0\leqslant i\leqslant n-1$*. We define the Goppa code over* $\mathbb{F}_q$ *associated to* $g(z)$ *as the code* $\mathscr{G}_q(\mathbf{x},g(z)) \stackrel{def}{=} \mathcal{A}_t(\mathbf{x},\mathbf{y})$*, with* $\mathbf{y} = g(\mathbf{x})^{-1}$*. The dimension* k *of* $\mathscr{G}_q(\mathbf{x},g(z))$ *satisfies* $k\geqslant n-tm$*. The polynomial* $g(z)$ *is called the* Goppa polynomial*, and* m *is the* extension degree*. Equivalently,* $\mathscr{G}_q\big(\mathbf{x},g(z)\big)$ *can be defined as:*

$$\mathscr{G}_q\big(\mathbf{x},g(z)\big) \stackrel{def}{=} \left\{\mathbf{c}=(c_0,\ldots,c_{n-1})\in\mathbb{F}_q^n \mid \sum_{i=0}^{n-1}\frac{c_i}{z-x_i} \equiv 0 \bmod g(z)\right\}.$$

Goppa codes naturally inherit a decoding algorithm that corrects up to $\frac{t}{2}$ errors. This bound can be improved to correct more errors by using *Wild* Goppa codes, introduced by Bernstein, Lange, and Peters in [6]. We also recall the version of Wild Goppa code used in Wild McEliece Incognito [8]. We call such special version of Wild Goppa codes: *Masked Wild Goppa codes.*

Definition 3 (Wild Goppa/Masked Wild Goppa). *Let* $\mathbf{x}$ *be an* n*-tuple* $(x_0,\ldots,x_{n-1})$ *of distinct elements of* $\mathbb{F}_{q^m}$*. Let* $\Gamma(z) \in \mathbb{F}_{q^m}[z]$ *(resp.* $f(z) \in$

$\mathbb{F}_{q^m}[z]$*) be a squarefree polynomial of degree* t *(resp.* u*) satisfying* $\Gamma(x_i) \neq 0$ *(resp.* $f(x_i) \neq 0$*) for all* $i, 0 \leqslant i \leqslant n-1$*. A* Wild Goppa code *is a Goppa code whose Goppa polynomial is of the form* $g(z) = \Gamma(z)^{q-1}$*. A* Masked Wild Goppa code *is a Wild Goppa code whose Goppa polynomial is such that* $g(z) = f(z)\Gamma(z)^{q-1}$*.*

The reason for using those Goppa polynomials lies in the following result.

Theorem 1. *[6,8] Let the notations be as in Definition 3. It holds that*

$$\mathscr{G}_q\big(\mathbf{x}, f(z)\Gamma^{q-1}(z)\big) = \mathscr{G}_q\big(\mathbf{x}, f(z)\Gamma^{q}(z)\big). \tag{3}$$

Thus, the code $\mathscr{G}_q\big(\mathbf{x}, f(z)\Gamma^{q}(z)\big)$ *has dimension* $\geqslant n - m\,((q-1)t + u)$*.*

This is a generalization of a well-known property for $q = 2$. The advantage of Wild Goppa codes (i.e. $f = 1$) compared to standard Goppa codes is that $\lfloor qt/2 \rfloor$ errors can be decoded efficiently (instead of $\lfloor (q-1)t/2 \rfloor$) for the same code dimension ($n - (q-1)mt$ in most cases). In fact, we can decode up to $\lfloor qt/2 \rfloor + 2$ using list decoding. This increases the difficulty of the syndrome decoding problem. Hence, for a given level of security, codes with smaller keys can be used (for details, see [6, Section 7] and [8, Section 5]).

3 An Algebraic Modelling with a Vector Space Structure on the Zero Set

The core idea of our attack is to construct, thanks to the public matrix, an algebraic system whose solution set $\mathcal{S}$ has a very surprising structure (Definition 4). It appears that $\mathcal{S}$ includes the union of several vector spaces. The vector spaces correspond in fact to sums of GRS codes (Definition 1) which have almost the same support $\mathbf{x}$ and multiplier vector $\mathbf{y}$ as the public-key of the attacked Wild McEliece Incognito scheme (Theorem 3). These vectors give a key-equivalent to the secret-key.

3.1 Description of the New Modelling

We consider the following algebraic equations:

Definition 4. *Let* $q = p^s$ *(*p *prime and* $s \geqslant 0$*). Let* $\mathbf{G}_{pub} = (g_{i,j})_{\substack{0 \leqslant i \leqslant n-1 \\ 0 \leqslant j \leqslant k-1}}$ *be a generator matrix of a masked Wild Goppa code* $\mathscr{C}_{pub} = \mathscr{G}_q(\mathbf{x}, f(z)\Gamma^{q-1}(z))$*. For an integer* $a, 0 < a \leqslant s$*, we define the system* $\mathcal{W}_{q,a}(\mathbf{Z})$ *as follows :*

$$\mathcal{W}_{q,a}(\mathbf{Z}) = \bigcup_{u \in \mathcal{P}_a} \left\{ \sum_{j=0}^{n-1} g_{i,j} Z_j^u = 0 \quad | \quad 0 \leqslant i \leqslant k-1 \right\} \tag{4}$$

with $\mathcal{P}_a = \{1, 2, \ldots, p^a - 1\} \cup \{p^a, p^{a+1}, \ldots, q\}$*.*

The parameter a in $\mathcal{P}_a$ determines the exponents considered for the Z_j's in the system (4). For $a = s$, we consider all the powers Z_j^u where u ranges in $\{1, \ldots, q\}$. Removing some exponents leads to a system with fewer equations and may seem counter-intuitive at first sight (the more equations, the better it is for solving a polynomial system). However, the situation is different here due to the specific structure of the solutions of $\mathcal{W}_{q,a}(\mathbf{Z})$, described in the following theorem.

Theorem 2. *Let the notations be as in Definition 4. Let* $\mathbf{y} = \Gamma(\mathbf{x})^{-1}$, $t = \deg(\Gamma)$ *and* $\mathcal{L}_a = \bigcup_{0 \leqslant r \leqslant s-1-a} \{p^r, 2p^r, \ldots, (p-1)p^r\} \cup \{p^{s-a}\}$. *The solutions* $\mathcal{S}$ *of* $\mathcal{W}_{q,a}(\mathbf{Z})$ *contain the union of* m *vector spaces which are sums of* GRS *codes:*

$$\bigcup_{0 \leqslant e \leqslant m-1} \left(\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e} \right) \subseteq \mathcal{S},$$

with $\mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e}$ *denoting all the elements of* $\mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)$ *with coordinates raised to the power* q^e, *with* $0 \leqslant e \leqslant m-1$.

Remark 1. When all the powers $\{1, \ldots, q\}$ are considered in the system, that is $a = s$, then $\mathcal{L}_a$ is reduced to $\{1\}$ and the solution set is a union of GRS codes. If $a < s$, the solution set is a bit more complex, but it has the great advantage of having a larger dimension; allowing then to solve the system (4) more efficiently. We will formalize this in Section 3.2.

Note that we state in Theorem 2 that we know a subset of the solutions. In practice, as the system is highly overdefined, we always observed that this subset was *all* the solutions.

Proof. The full proof of this result is postponed in Section A.3. We just give the global idea of the proof. The goal is to show the elements of $\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e}$ are solutions of $\mathcal{W}_{q,a}(\mathbf{Z})$. We can assume that $e = 0$ w.lo.g.

Let $\mathbf{z} = (z_1, \ldots, z_n) \in \sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)$. We write the coordinates of $\mathbf{z}$ as $z_j = \sum_{\ell \in \mathcal{L}_a} y_j^\ell Q_\ell(x_j^\ell)$, where the Q_ℓ's are polynomials of degree $\leqslant t-1$ of $\mathbb{F}_{q^m}[z]$. We have to prove that

$$\sum_{j=0}^{n-1} g_{i,j} z_j^u = 0 \text{ for } u \in \mathcal{P}_1 \cup \mathcal{P}_2, \text{where } \mathcal{P}_1 = \{1, 2, \ldots, p^a - 1\},\ \mathcal{P}_2 = \{p^a, \ldots, p^s\}.$$

The idea is to develop $z_j^u = \left(\sum_{\ell \in \mathcal{L}_a} y_j^\ell Q_\ell(x_j^\ell) \right)^u$ with Newton multinomial. The development is performed slightly differently whether $u \in \mathcal{P}_1$ or $u \in \mathcal{P}_2$ (see Appendix A.3). In both cases, we end up with a result of the form $z_j^u = \sum_{u_x, u_y} \alpha_{u_x,u_y} y_j^{u_y} x_j^{u_x}$, so that our sum writes:

$$\sum_{j=0}^{n-1} g_{i,j} z_j^u = \sum_{u_x, u_y} \left(\alpha_{u_x,u_y} \sum_{j=0}^{n-1} g_{i,j} y_j^{u_y} x_j^{u_x} \right). \tag{5}$$

Then, we apply the next lemma (proved in Appendix A.2).

Lemma 3 *Let* $\mathbf{G}_{pub}$ *be a generator matrix of a masked Wild Goppa code* $\mathscr{C}_{pub} = \mathscr{G}_q(\mathbf{x}, f(z)\Gamma^{q-1}(z)), \mathbf{y} = \Gamma(\mathbf{x})^{-1}$, $\mathbf{w} = f(\mathbf{x})^{-1}$ *and* $t = \deg(\Gamma(z)$. *The values of* $\mathbf{x}$, $\mathbf{y}$, *and* $\mathbf{w}$ *satisfy the following set of equations for any value of* u_x, u_y, u, b *verifying the conditions* $0 \leqslant u_y \leqslant q, 0 \leqslant u_x \leqslant u_y t - 1, 0 \leqslant u \leqslant \deg(f) - 1, b \in \{0, 1\}$ *and* $(b, u_y) \neq (0, 0)$*:*

$$\left\{ \sum_{j=0}^{n-1} g_{i,j} \left(w_i x_i^u\right)^b y_j^{u_y} x_j^{u_x} = 0 \mid 0 \leqslant i \leqslant k-1 \right\}.$$

We set $b = 0$ and obtain that $\sum_{j=0}^{n-1} g_{i,j} y_j^{u_y} x_j^{u_x} = 0$ for (u_y, u_x) such that $1 \leqslant u_y \leqslant t$ and $0 \leqslant u_x \leqslant u_y t - 1$. Thus to conclude that $\sum_{j=0}^{n-1} g_{i,j} z_j^u = 0$, we check that all the couples (u_x, u_y) appearing in the sum (5) satisfy those conditions.

3.2 Recovering a Basis of the Vector Subspace

We now explain more precisely how to use the particular structure of the solution set for solving the non-linear system (4). When looking for a vector in a subspace of $\mathbb{F}_{q^m}^n$ of dimension d, then you can safely fix d coordinates arbitrarily and complete the $n - d$ so as to obtain a vector of this subspace. This corresponds to computing intersections of your subspace with d hyperplanes. With this idea, we deduce the following corollary of Theorem 2.

Corollary 4 *Let $\mathscr{C}_{pub} = \mathscr{G}_q(\mathbf{x}, f(z)\Gamma^{q-1}(z))$ be a masked Wild Goppa code. Let $t = \deg(\Gamma)$, $\mathcal{W}_{q,a}(\mathbf{Z})$, and $\mathcal{L}_a$ be as defined in Theorem 2. Then, we can fix $t \times \#\mathcal{L}_a$ variables $\mathbf{Z}_i$ to arbitrary values in $\mathcal{W}_{q,a}(\mathbf{Z})$. The system obtained has m solutions (counted without multiplicities), one for each sum $\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e}$.*

In the rest of this article, we set $\lambda_{a,t} = t \times \#\mathcal{L}_a$. Our purpose is to find a basis of one of the vector spaces $\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e}$. To do so, we pick $\lambda_{a,t}$ independent solutions of $\mathcal{W}_{q,a}(\mathbf{Z})$ by fixing the variables $Z_0, Z_1, \ldots, Z_{\lambda_{a,t}-1}$ in $\mathcal{W}_{q,a,t}(\mathbf{Z})$ accordingly. Namely, for $0 \leqslant i \leqslant \lambda_{a,t} - 1$, we pick one solution $\mathbf{v}^{(i)}$ among the m solutions of the system

$$\mathcal{W}_{q,a}(\mathbf{Z}) \bigcup \{Z_i = 1, Z_j = 0 \mid 0 \leqslant j \neq i \leqslant \lambda_{a,t} - 1\}.$$

Thanks to Theorem 3 and Definition 1, we know that those solutions can be written as follows, for $Q_{i,\ell} \in \mathbb{F}_{q^m}[z]$ of degree lower than t and $0 \leqslant e_i \leqslant m - 1$:

$$\mathbf{v}^{(i)} = \left(0, \ldots, 1, \ldots, 0, \sum_{\ell \in \mathcal{L}_a} y_{\lambda_{a,t}}^{q^{e_i}} Q_{i,\ell}(x_{\lambda_{a,t}}^{qe_i}), \ldots\right) \in \sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^{e_i}}. \quad (6)$$

After $\lambda_{a,t}$ resolutions of $\mathcal{W}_{q,a}(\mathbf{Z})$, the solutions $\mathbf{v}^{(i)}$ are not necessarily a basis of one of the vector spaces $\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e}$ because the Fröbenius exponents need not be identical for all $\mathbf{v}^{(i)}$'s. We explain in the next paragraph why this is not an issue in practice.

Simplication: Fröbenius Alignment. Let $\{\mathbf{v}^{(i)}\}_{0 \leqslant i \leqslant \lambda_{a,t}-1}$ be as defined in (6). We can suppose without loss of generality that $q_0 = q_1 = \ldots = q_{\lambda_{a,t}-1}$. This simplification requires less than $m^{(\lambda_{a,t}-1)}$ Fröbenius evaluations on the solutions. Indeed, $\mathbf{v} \in \sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^e}$, implies that $\mathbf{v}^q \in \sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)^{q^{e+1}}$. For the parameters considered in [6,8], m and t are rather small, making the cost of the Fröbenius alignment negligible. In the rest of this article, we assume that $q_0 = \ldots = q_{\lambda_{a,t}-1} = 0$, which is not a stronger assumption since the private elements of $\mathscr{C}_{pub}$ are already defined up to Fröbenius endomorphism.

Example 1. Pick for instance $q = 8$ and solve the system $\mathcal{W}_{q,a}$ with $a = 2$. Thanks to Theorem 2, after re-alignment of the Fröbenius exponents, we have a basis of the vector space $\mathrm{GRS}_t(\mathbf{x}, \mathbf{y}) + \mathrm{GRS}_t(\mathbf{x}^2, \mathbf{y}^2)$, that is:

$$\left\{(y_i Q(x_i) + y_i^2 R(x_i^2))_{0 \leqslant i \leqslant n-1} | Q, R \in \mathbb{F}_{q^m}[z], \deg(Q), \deg(R) \leqslant t - 1\right\}.$$

4 Recovering the Secret Key from a Sum of GRS – A Linear Algebra Step

Once we know a basis $(\mathbf{v}^{(i)})_{0\leqslant i\leqslant \lambda_{a,t}-1}$ of $\sum_{\ell\in\mathcal{L}_a}\mathrm{GRS}_t(\mathbf{x}^\ell,\mathbf{y}^\ell)$, we aim at recovering the basis of a single GRS code. This *disentanglement* is done in Paragraph 4.1. Then, we show in 4.2 how to recover a private support $\mathbf{x}$ and Goppa polynomial $\Gamma(z)$ of the masked Wild Goppa code. This is the full description of a plain Wild Goppa code. In the Incognito case ($\deg(f)>0$), we explain in 4.3 that an extra linear step enables to find f. To sum up, the purpose of this section is to prove the following theorem.

Theorem 5. *Let $q=p^s$ (p prime and $s\geqslant 0$). Let $\mathbf{G}_{pub}=(g_{i,j})_{\substack{0\leqslant i\leqslant n-1\\0\leqslant j\leqslant k-1}}$ be a generator matrix of a masked Wild Goppa code $\mathscr{C}_{pub}=\mathscr{G}_q(\mathbf{x},f(z)\Gamma^{q-1}(z))$. Let $\mathbf{y}=\Gamma(\mathbf{x})^{-1}$, $t=\deg(\Gamma)$, and*

$$\mathcal{V}=\left(\sum_{\ell\in\mathcal{L}_a}\mathrm{GRS}_t(\mathbf{x}^\ell,\mathbf{y}^\ell)\right)\ \text{where}\ \mathcal{L}_a=\bigcup_{r=0}^{s-1-a}\{p^r,2p^r,\ldots,(p-1)p^r\}\cup\left\{p^{s-a}\right\}.$$

Once $\mathcal{V}$ is given, we can recover in polynomial-time a support $\mathbf{x}'$ and polynomials $f'(z),\Gamma'(z)\in\mathbb{F}_{q^m}[z]$ such that $\mathscr{C}_{pub}=\mathscr{G}_q(\mathbf{x}',f'(\Gamma')^{q-1})$. Stated differently, we can recover in polynomial-time a key $(\mathbf{x}',\Gamma',f')$ equivalent to the secret-key as soon as the system (4) has been solved.

4.1 Disentanglement of the System Solutions

The Sidelnikov-Shestakov [26] attack is a well known attack against McEliece schemes instantiated with GRS codes [22]. In our case, we can have a sum of GRS codes. In this situation, it seems not possible to apply directly [26] (because the vectors of $\sum_{\ell\in\mathcal{L}_a}\mathrm{GRS}_t(\mathbf{x}^\ell,\mathbf{y}^\ell)$ do not have the desired form; that is $(y_0Q(x_0),\ldots,y_{n-1}Q(x_{n-1}))$. To overcome this issue, we propose to use well-chosen intersections to recover a basis suitable for Sidelnikov-Shestakov. To gain intuition, we provide a small example.

Example 2. We continue with the example 1. By squaring all the elements of $\mathrm{GRS}_t(\mathbf{x},\mathbf{y})+\mathrm{GRS}_t(\mathbf{x}^2,\mathbf{y}^2)$, we have a basis of $\mathrm{GRS}_t(\mathbf{x}^2,\mathbf{y}^2)+\mathrm{GRS}_t(\mathbf{x}^4,\mathbf{y}^4)$:

$$\left\{(y_i^2Q(x_i^2)+y_i^4R(x_i^4))_{0\leqslant i\leqslant n-1}|Q,R\in\mathbb{F}_{q^m}[z],\deg(Q),\deg(R)\leqslant t-1\right\}.$$

We prove in Proposition 6 that, in charac. 2,

$$\big(\mathrm{GRS}_t(\mathbf{x},\mathbf{y})+\mathrm{GRS}_t(\mathbf{x}^2,\mathbf{y}^2)\big)\cap\big(\mathrm{GRS}_t(\mathbf{x}^2,\mathbf{y}^2)+\mathrm{GRS}_t(\mathbf{x}^4,\mathbf{y}^4)\big)=\mathrm{GRS}_t(\mathbf{x}^2,\mathbf{y}^2).$$

Hence, we have a basis of $\mathrm{GRS}_t(\mathbf{x}^2,\mathbf{y}^2)$.

Our general method to disentangle the solutions is proved in characteristic 2, but for other characteristics we need the following assumption:

Assumption 1 *Let $q = p^s$ with p prime. Let $\mathbf{x} \in \mathbb{F}_{q^m}^n$ be a support and $\mathbf{y} \in \mathbb{F}_{q^m}^n$ be defined by $\mathbf{y} = \Gamma(\mathbf{x})^{-1}$ for some polynomial $\Gamma(z) \in \mathbb{F}_{q^m}[z]$ of degree t. Let $\mathcal{L}$ and $\mathcal{L}'$ be two subsets of $\{1, \ldots, q\}$ with $(\#\mathcal{L} + \#\mathcal{L}')t < n$. Then, we have that:*

$$\left(\sum_{\ell \in \mathcal{L}} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)\right) \bigcap \left(\sum_{\ell \in \mathcal{L}'} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)\right) = \sum_{\ell \in \mathcal{L} \cap \mathcal{L}'} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell).$$

For the specific subsets $\mathcal{L}$ that we encountered, this assumption is rigorously proved in characteristic 2 (see Proposition 6). For bigger characteristics, though we could not find a formal proof, we launched more than $100,000$ experiments and found out that equality held in all cases. Now we generalize the method of intersection of codes proposed in Example 2.

Proposition 6 *Let $q = p^s$ (p prime and $s \geqslant 0$). Let also $a, 0 < a \leqslant s$, and $\mathcal{L}_a = \bigcup_{0 \leqslant r \leqslant s-1-a} \{p^r, 2p^r, \ldots, (p-1)p^r\} \cup \{p^{s-a}\}$. Then:*

$$\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell) \cap \left(\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)\right)^{p^{(s-a)}} = \mathrm{GRS}_t(\mathbf{x}^{p^{s-a}}, \mathbf{y}^{p^{s-a}}).$$

Proof. Let $\Phi : (m_0, \ldots, m_{n-1}) \in \mathbb{F}_{q^m}^n \mapsto (m_0^{p^{s-a}}, \ldots, m_{n-1}^{p^{s-a}})$. First, remark that, as p^{s-a} is a power of the characteristic, it holds that $\Phi\left(\mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)\right) = \mathrm{GRS}_t(\mathbf{x}^{p^{s-a}\ell}, \mathbf{y}^{p^{s-a}\ell})$ for all ℓ, and $\Phi\left(\sum_{\ell \in \mathcal{L}_a} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)\right) = \sum_{\ell \in \Phi(\mathcal{L}_a)} \mathrm{GRS}_t(\mathbf{x}^\ell, \mathbf{y}^\ell)$. When $p = 2$, we fully prove the proposition in Appendix A.4. Otherwise (when $p > 2$), we rely on Assumption 1 with the sets $\mathcal{L}_a$ and

$$\Phi(\mathcal{L}_a) = \bigcup_{s-a \leqslant r \leqslant 2(s-a)-1} \{p^r, 2p^r, \ldots, (p-1)p^r\} \cup \left\{p^{2(s-a)}\right\}.$$

Then, we have $\mathcal{L}_a \cap \Phi(\mathcal{L}_a) = \{p^{s-a}\}$, and the desired equality. □

Once a basis of $\mathrm{GRS}_t(\mathbf{x}^{p^{s-a}}, \mathbf{y}^{p^{s-a}})$ is known, we recover $\mathbf{x}, \mathbf{y}$ and $\Gamma(z)$ thanks to a variant of Sidelnikov-Shestakov described below.

4.2 Sidelnikov-Shestakov Adapted To Recover the Goppa Polynomial

In our attack, we have to adapt the classical Sidelnikov-Shestakov attack for special GRS codes, namely those for which there is an additional polynomial relation between the support and the multipliers.

Proposition 7 *Let $\mathbf{x}$ be an n-tuple $(x_0, \ldots, x_{n-1})$ of distinct elements of $\mathbb{F}_{q^m}$ and $\Gamma(z) \in \mathbb{F}_{q^m}[z]$ be a squarefree polynomial of degree t such that $\Gamma(x_i) \neq 0$, for all $i, 0 \leqslant i \leqslant n-1$ Let $\mathbf{G}_{\mathrm{GRS}}$ be the generator matrix of a* GRS *code $\mathrm{GRS}_t\big(\mathbf{x}, \Gamma(\mathbf{x})^{-1}\big)$. There is a polynomial-time algorithm which allows to recover a n-tuple $\mathbf{x}' = (x'_0, \ldots, x'_{n-1})$ of distinct elements of $\mathbb{F}_{q^m}$ and a squarefree polynomial $\Gamma'(z) \in \mathbb{F}_{q^m}[z]$ of degree t such that $\Gamma'(x'_i) \neq 0$, for all $i, 0 \leqslant i \leqslant n-1$ and $\mathrm{GRS}_t\big(\mathbf{x}, \Gamma(\mathbf{x})^{-1}\big) = \mathrm{GRS}_t\big(\mathbf{x}', \Gamma'(\mathbf{x}')^{-1}\big)$.*

This problem is very close to the one addressed in [26]. The only issue is that the homographic transformation on the support used in the original attack indeed preserves the GRS structure but not the polynomial link. Thus, polynomial interpolation over $\mathbf{x}$ and $\mathbf{y}^{-1}$ is not possible. We propose to avoid this homographic transformation by considering a well chosen *extended code*.

Definition 5. *Let $\mathscr{C}$ be a linear code of length n over $\mathbb{F}_q$. The extended code of $\mathscr{C}$, denoted by $\widehat{\mathscr{C}}$, is a code of length $n+1$ obtained by adding to each codeword $\mathbf{m} = (m_0, \ldots, m_{n-1})$ the coordinate $-\sum_{j=0}^{n-1} m_j$.*

Our algorithm, proved in the full version of this paper , is then the following.

Algorithm 1 Extended Version of Sidelnikov-Shestakov algorithm

INPUT : $\mathbf{G}_{GRS}$ generator matrix of $\mathscr{C}_{GRS} = \mathrm{GRS}_t(\mathbf{x}, \mathbf{y})$, with $\mathbf{y} = \Gamma(\mathbf{x})^{-1}$ ($\deg(\Gamma) = t$)
OUTPUT : Secret $\mathbf{x}$, $\mathbf{y}$, and $\Gamma(z)$

1: Build $\mathbf{P} = (p_{i,j})_{\substack{0 \leqslant i \leqslant n-t-1 \\ 0 \leqslant j \leqslant n-1}}$ a generator matrix of the dual of $\mathscr{C}_{GRS}$.
2: Deduce $\tilde{\mathbf{P}}$ a matrix of the extended code (Definition 5) of the code spanned by $\mathbf{P}$
3: Build $\left(\mathbf{I}_t | \mathbf{U}\right)$, with $\mathbf{U} = (u_{i,j})_{\substack{0 \leqslant i \leqslant t \\ t+1 \leqslant j \leqslant n}}$ a parity-check matrix of the code spanned by $\tilde{\mathbf{P}}$ in systematic form
4: Solve the linear system with unknowns X_i's to find $\mathbf{x}$

$$\left\{ \frac{u_{i,j}}{u_{i',j}}(X_{i'} - X_j) = \frac{u_{i,n}}{u_{i',n}}(X_i - X_j) \mid 0 \leqslant i, i' \leqslant t, t+1 \leqslant j \leqslant n-1 \right\}.$$

5: Solve the linear system with unknowns Y_i's to find $\mathbf{y}$ (the x_i's were found at previous step)

$$\left\{ \sum_{i=0}^{n-1} p_{j,i} x_i^{\ell} Y_i = 0 \mid 0 \leqslant j \leqslant n-t-1, 0 \leqslant \ell \leqslant t-1 \right\}.$$

6: Interpolate $\Gamma(z)$ from $\mathbf{x}$ and $\mathbf{y}^{-1}$

4.3 Recovery of the Incognito Polynomial by Solving a Linear System

An extra step is necessary in the Incognito case to recover the other factor f of the Goppa polynomial. To do so, we recover the multipliers associated to f, that is the vector $\mathbf{w} = f(\mathbf{x})^{-1}$. Then, we perform polynomial interpolation. We note that once $\mathbf{x}$ and $\mathbf{y} = \Gamma(\mathbf{x})^{-1}$ are known, many of the equations of Lemma 3 become linear in $\mathbf{w}$. Namely,

$$\bigcup_{u_y=1}^{q} \left\{ \sum_{j=0}^{n-1} g_{i,j} w_i \left(y_j^{u_y} x_j^{u_x} \right) = 0 \mid 0 \leqslant i \leqslant k-1, 0 \leqslant u_x \leqslant u_y t + \deg(f) - 1 \right\}.$$

In practice, we observed that the linear system obtained has a rank defect and is not sufficient to find $\mathbf{w}$. However, we can also use the fact that $\mathscr{C}_{pub} \subset \mathscr{G}(\mathbf{x}, f(z))$ to prove that

$$\sum_{j=0}^{n-1} g_{i,j} w_i x_i^{\deg(f)} = \frac{1}{\mathrm{LC}(f)} \left(\sum_{j=0}^{n-1} g_{i,j} \right).$$

(This is rigorously done in the full version of this article.) Since $\mathbf{x}$ is known and setting $\mathrm{LC}(f) = 1$, we obtain new linear equations in the components of $\mathbf{w}$. Putting all the linear equations together, experiments show then that we obtain a unique solution $\mathbf{w}$, and f by polynomial interpolation.

5 Weakness of Non-Prime Base Fields

The most (computationally) difficult part of our attack against Wild McEliece Incognito is to solve the algebraic system defined in Theorem 2. In this part, we aim at giving a better idea of the complexity of resolution by determining the exact number of "free variables" in the system. Namely, we show that we can eliminate many variables thanks to linear equations. The system $\mathcal{W}_{q,a}(\mathbf{Z}) = \bigcup_{u \in \mathcal{P}_a} \left\{ \sum_{j=0}^{n-1} g_{i,j} Z_j^u = 0 \quad | \quad 0 \leqslant i \leqslant k-1 \right\}$ of Theorem 2 obviously contains k linear equations by picking $u = 1$ ($1 \in \mathcal{P}_a$ by definition). We can easily derive other linear equations by applying the additive map $z \mapsto z^{(q^m/p^u)}$ to all the equations in degree p^u. As the solutions lie in $\mathbb{F}_{q^m}$, it holds that $(Z_j^{p^u})^{q^m/p^u} = Z_j$, and for $0 \leqslant i \leqslant k-1$

$$\left(\sum_{j=0}^{n-1} g_{i,j} Z_j^{p^u} \right)^{q^m/p^u} = \sum_{j=0}^{n-1} g_{i,j}^{q^m/p^u} Z_j = 0.$$

However, we observed that those linear equations were very redundant. To explain those linear dependencies, we found out a property of the masked Wild Goppa codes $\mathscr{G}_q(\mathbf{x}, f(z)\Gamma(z)^q)$ (Theorem 8). Namely, by simple operations on their generator matrices, we can build a generator matrix of the code $\mathscr{G}_p(\mathbf{x}, \Gamma(z)^p)$ over $\mathbb{F}_p$. This latter matrix allows to write many *independent* linear equations implying the private elements of $\mathscr{C}_{pub}$.

Theorem 8. *Let $q = p^s$ (p prime, and $s > 0$). Let $\mathbf{G}_{pub} = (g_{i,j})_{\substack{0 \leqslant i \leqslant n-1 \\ 0 \leqslant j \leqslant k-1}}$ be a generator matrix of a masked Wild Goppa code $\mathscr{C}_{pub} = \mathscr{G}_q(\mathbf{x}, f(z)\Gamma(z)^{q-1})$. We consider the scalar restriction of $\mathbf{m} \in \mathscr{C}_{pub} \subseteq \mathbb{F}_q^n$ into $\mathbb{F}_p^s$. This yields s components $\mathbf{m}^{(0)}, \ldots, \mathbf{m}^{(s-1)} \in \mathbb{F}_p^n$ (we write each $\mathbf{m} \in \mathbb{F}_q^n$ over a $\mathbb{F}_p$-basis, i.e. $\mathbf{m} = \mathbf{m}^{(0)}\theta_0 + \cdots + \mathbf{m}^{(s-1)}\theta_{s-1}$). Let $\mathscr{C}^{\mathbb{F}_p} \subseteq \mathbb{F}_p^n$ be the code generated by the coordinate vectors $\mathbf{m}^{(0)}, \ldots, \mathbf{m}^{(s-1)}$ for all the codewords $\mathbf{m} \in \mathscr{C}_{pub}$. Then, it holds that*

$$\mathscr{C}^{\mathbb{F}_p} \subseteq \mathscr{G}_p(\mathbf{x}, \Gamma(z)^p).$$

The proof can be found in the full version of this paper.In practice, we observed equality in the inclusion provided $s \dim(\mathscr{C}_{pub}) > \dim(\mathscr{G}_p(\mathbf{x}, \Gamma(z)^p))$. Note that $\mathscr{G}_p(\mathbf{x}, \Gamma(z)^p)$ is a Wild Goppa code with the same private elements $\mathbf{x}$ and $\mathbf{y} = \Gamma(\mathbf{x})^{-1}$ as $\mathscr{C}_{pub}$. This provides extra equations on the variables $\mathbf{Z}$ of $\mathcal{W}_{q,a}(\mathbf{Z})$ (proved in the full version):

Proposition 9 *Let $\mathscr{C}_{pub} = \mathscr{G}_q(\mathbf{x}, f(z)\Gamma^{q-1}(z))$ and $\mathcal{W}_{q,a}(\mathbf{Z})$ the associated system for $1 \leqslant a \leqslant s$. Let $\widetilde{\mathbf{G}}_{\mathbb{F}_p} = (\widetilde{g}_{i,j})_{\substack{0 \leqslant i \leqslant n-1 \\ 0 \leqslant j \leqslant k_p - 1}}$ be a generator matrix of $\mathscr{G}_p(\mathbf{x}, \Gamma(z)^p)$ (with $k_p = \dim(\mathscr{G}_p(\mathbf{x}, \Gamma(z)^p))$). Then, the solutions of $\mathcal{W}_{q,a}(\mathbf{Z})$ satisfy:*

$$\bigcup_{\ell=0}^{p-1} \left\{ \sum_{j=0}^{n-1} \widetilde{g}_{i,j} Z_j = 0 \mid 0 \leqslant u \leqslant t-1, 0 \leqslant i \leqslant k_p - 1 \right\}.$$

As $k_p \geqslant n - (p-1)mst$ (and in practice $k_p = n - (p-1)mst$), we have the following corollary.

Corollary 10 *The knowledge of $\mathbf{G}_{pub}$ gives access to $n-(p-1)mst$ independent linear relations between the Z_i's. The system $\mathcal{W}_{q,a}(\mathbf{Z})$ contains (at most) $(p-1)mst$ free variables.*

Remark 2. The number of "free" variables given in Corollary 10 is given without taking into account the vector space structure of the solutions. Thanks to Corollary 4, we know that $\lambda_{a,t}$ extra variables can be fixed to arbitrary values in $\mathcal{W}_{q,a}(\mathbf{Z})$.

For a Goppa polynomial of same degree, but without multiplicities, the number of free variables in the system would be $n-k \geqslant (p^s-1)mt$ instead of $(p-1)mst$. In particular, for a masked code, the number of variables describing it does not depend on the degree of the incognito polynomial f and the attack is not harder for masked codes. This explains why the codes defined over non-prime fields are the weakest ones.

6 Practical Experiments

We report below various experimental results performed with our attack on various parameters for which [6] said that strength is "unclear" and that an attack would not be a "surprise" but for which no actual attack was known. We also generated our own keys/parameters to see how the attack scales. We performed our experiments with off-the-shelf tools (Magma [9] V2.19-1) and using a 2.93 GHz Intel PC with 128 Gb. of RAM. As the polynomial system solving is by far the most costly step, we give timings only for this one. We performed it using the F_4 algorithm ([12]) of Magma. As explained in Section 4, it is necessary to solve the systems $\mathcal{W}_{q,a}(\mathbf{Z})$ a number of times equal to the dimension of the vector space of the solutions (Theorem 2). These resolutions are completely independent and can be executed in parallel. This is why we give the timings under the form (number of separate resolutions) $\times$ (time for one resolution). By $\#\mathbf{Z}$, we denote the number of free variables remaining in the system after cleaning up the linear equations (Corollary 10) and fixing coordinates thanks to the vector space structure of the solutions (Corollary 4). The general formula is $\#\mathbf{Z} = ((p-1)ms - \#\mathcal{L}_a)\,t$ for $q = p^s$ and $s > 1$.

In the experiments, we tried various parameters a for the systems $\mathcal{W}_{q,a}(\mathbf{Z})$. We give a comparison on some examples in Table 1 (the system $\mathcal{W}_{q,a}(\mathbf{Z})$ with $a = s$ can be solved in a reasonable amount of time in actually few cases).

q	m	t	n	k	$\deg(f)$	Solving $\mathcal{W}_{q,a}(\mathbf{Z})$ with $a=s$	Solving $\mathcal{W}_{q,a}(\mathbf{Z})$, **optimal** a
32	2	2	678	554	0	$2\times$ 12s ($\#\mathbf{Z}=18$)	$8\times$ 0.08 s ($a=2, \#\mathbf{Z}=9$)
32	2	1	532	406	32	$2\times$ 49s ($\#\mathbf{Z}=9$)	$4\times$ 0.02 s ($a=2, \#\mathbf{Z}=6$)
32	2	3	852	621	24	$3\times$ (30 min 46s) ($\#\mathbf{Z}=37$)	$12\times$ 0.6 s ($a=2, \#\mathbf{Z}=18$)
27	3	3	1312	1078	0	$3\times$ (3h 10 min) ($\#\mathbf{Z}=51$)	$15\times$ 3.0 s ($a=1, \#\mathbf{Z}=39$)

Table 1. Comparison of the resolution times of $\mathcal{W}_{q,a}(\mathbf{Z})$ for various possible a's. The smallest possible a gives the best timings.

It appeared that a should be chosen so as to maximize the dimension of the solution set (Theorem 2). This choice minimizes the number of variables. Namely, the best choice is to set $a=1$ when $p>2$. When $p=2$, setting $a=1$ would yield only "linear" equations (of degree $2^u, u \leqslant s$). So, we set $a=2$ and the systems $\mathcal{W}_{2^u,2}(\mathbf{Z})$ contain only cubic equations. We recall that for $a=s$, Assumption 1 is not necessary, whereas we rely on it when $a<s$ and $p\neq 2$. In the rest of the experiments, we always pick the best choice for a.

In Table 2, we present experimental results performed with Wild McEliece (when $\deg(f)=0$) and Incognito ($\deg(f)>0$) parameters. For Wild McEliece, all the parameters in the scope of our attack were quoted in [6, Table 7.1] with the international biohazard symbol ☣. The reason is that, for those parameters, enumerating all the possible Goppa polynomials is computationaly feasible. In the current state of the art, to apply the SSA attack ([18]), one would not only have to enumerate the irreducible polynomials of $\mathbb{F}_{q^m}[z]$, but also all the possible support sets, as the support-splitting algorithm uses the support set as input. This introduces a factor $\binom{q^m}{n}$ in the cost of SSA, chosen by the designers in order to make the attack infeasible. However, the authors of [6] conclude that, even if no attack is known against those instances, algorithmical progress in support enumeration may be possible and therefor they do not recommend their use. In the case of non-prime base fields, experiments show that our attack represents a far more serious threat for the security of some of those instances: for $q \in \{32, 27, 16\}$ we could find the secret keys of parameters with high ISD complexity. We indicate, for each set of parameters, the ISD complexity (obtained thanks to Peters' software [1]), as it remains the reference to evaluate the security of a McEliece scheme. We also give the complexity of an SSA attack, which is in the current state-of-the-art $\binom{q^m}{n} \cdot q^{mt}/t$.

Regarding Wild McEliece Incognito, we broke the parameters indicated with a security of 2^{128} in [8, Table 5.1] for $q \in \{32, 27, 16\}$. For some other non-prime base fields, we give the hardest parameters in the scope of our attack in roughly one day of computation. Note that here, SSA complexity is given by $\binom{q^m}{n} \cdot (q^{m(t+s)}/(ts))$.

For the sake of completeness, we also include in Tables 2 Wild McEliece schemes with a quadratic extension. In [11], the authors already presented a polytime attack in this particular case: it applies for the parameters with $q=32$, but not for the other ones. We want to stress that our attack also works for $m=2$

[1] available at `http://christianepeters.wordpress.com/publications/tools/`

and any t ([11] does not work in the extreme case $t = 1$). Also, we emphasize that, whilst solving a non-linear system, our attack is actually faster than [11] in some cases. For $q = 32$ and $t = 4$, the attack of [11] requires 49.5 minutes (using a non-optimized MAGMA implementation according to the authors). We can mount our attack in several seconds with the techniques of this paper.

q	m	t	n	k	$\deg(f)$	Key (kB)	**ISD**	**SSA**	Solving $\mathcal{W}_{q,a}(\mathbf{Z})$, **optimal** a
32	2	4	841	601	0	92	$\mathbf{2^{128}}$	$2^{688} \cdot 2^{38}$	16× 10 s ($\#\mathbf{Z} = 36$)
32	2	5	800	505	0	93	$\mathbf{2^{136}}$	$2^{771} \cdot 2^{48}$	20× (2 min 45s) ($\#\mathbf{Z} = 40$)
27	3	3	1312	1078	0	45	$\mathbf{2^{113}}$	$2^{6947} \cdot 2^{41}$	15× 3.0 s ($\#\mathbf{Z} = 39$)
27	3	4	1407	1095	0	203	$\mathbf{2^{128}}$	$2^{7304} \cdot 2^{55}$	20× (6 min 34 s) ($\#\mathbf{Z} = 52$)
27	3	5	1700	1310	0	304	$\mathbf{2^{158}}$	$2^{8343} \cdot 2^{69}$	25× (1h 59 min) ($\#\mathbf{Z} = 65$)
27	3	5	1800	1410	0	327	$\mathbf{2^{160}}$	$2^{8679} \cdot 2^{69}$	25× (1h 37 min) ($\#\mathbf{Z} = 65$)
16	3	6	1316	1046	0	141	$\mathbf{2^{129}}$	$2^{3703} \cdot 2^{69}$	18× (36h 26 min) ($\#\mathbf{Z} = 54$)
32	2	3	852	621	24	90	$\mathbf{2^{130}}$	$2^{663} \cdot 2^{273}$	12× 0.6 s ($\#\mathbf{Z} = 18$)
27	3	2	1500	1218	42	204	$\mathbf{2^{128}}$	$2^{5253} \cdot 2^{225}$	10× 0.9 s ($\#\mathbf{Z} = 26$)
25	3	3	1206	915	25	155	$\mathbf{2^{117}}$	$2^{7643} \cdot 2^{632}$	15× (1h 2 min) ($\#\mathbf{Z} = 57$)
16	3	6	1328	1010	16	160	$\mathbf{2^{125}}$	$2^{3716} \cdot 2^{265}$	18× (36h 35 min) ($\#\mathbf{Z} = 54$)
9	3	6	728	542	14	40	$\mathbf{2^{81}}$	$2^{2759} \cdot 2^{191}$	18× (25h 13 min) ($\#\mathbf{Z} = 54$)

Table 2. Practical experiments with Wild McEliece & Incognito parameters. ISD complexity is obtained thanks to Peters' software[1]. SSA attack complexity is given under the form (support enumeration)·(Goppa polynomial enumeration).

Conclusion and Future Work. In practice, we could not solve (in less than two days) the algebraic systems involved when the number of free variables $\#\mathbf{Z}$ exceeds 65. We recall the relation $\#\mathbf{Z} = ((p-1)ms - \#\mathcal{L}_a)\,t$ (for $q = p^s$ and $s > 1$), which should help the designers to scale their parameters. An important remaining open question is to give a precise complexity estimates for the polynomial system solving phase in those cases.

Acknowledgements. This work was supported in part by the HPAC grant (ANR ANR-11-BS02-013) of the French National Research Agency. The authors would also like to thank (some of) the referees as well as PC chairs for their usefull comments on a preliminary version of this paper.

References

1. M. Barbier and P. S. L. M. Barreto. Key reduction of McEliece's cryptosystem using list decoding. In A. Kuleshov, V. Blinovsky, and A. Ephremides, editors, *2011 IEEE International Symposium on Information Theory Proceedings, ISIT 2011, St. Petersburg, Russia, July 31 - August 5, 2011*, pages 2681–2685. IEEE, 2011.
2. P. S. L. M. Barreto, R. Lindner, and R. Misoczki. Monoidic codes in cryptography. In Yang [27], pages 179–199.
3. A. Becker, A. Joux, A. May, and A. Meurer. Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT*, volume 7237 of *Lecture Notes in Computer Science*, pages 520–536. Springer, 2012.

4. T. P. Berger, P. Cayrel, P. Gaborit, and A. Otmani. Reducing key length of the McEliece cryptosystem. In B. Preneel, editor, *Progress in Cryptology - AFRICACRYPT 2009, Second International Conference on Cryptology in Africa, Gammarth, Tunisia, June 21-25, 2009. Proceedings*, volume 5580 of *Lecture Notes in Computer Science*, pages 77–97. Springer, 2009.
5. D. J. Bernstein, T. Lange, and C. Peters. Attacking and defending the McEliece cryptosystem. In J. Buchmann and J. Ding, editors, *Post-Quantum Cryptography, Second International Workshop, PQCrypto 2008, Cincinnati, OH, USA, October 17-19, 2008, Proceedings*, volume 5299 of *Lecture Notes in Computer Science*, pages 31–46. Springer, 2008.
6. D. J. Bernstein, T. Lange, and C. Peters. Wild McEliece. In A. Biryukov, G. Gong, and D. R. Stinson, editors, *Selected Areas in Cryptography - 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers*, volume 6544 of *Lecture Notes in Computer Science*, pages 143–158. Springer, 2010.
7. D. J. Bernstein, T. Lange, and C. Peters. Smaller decoding exponents: Ball-collision decoding. In P. Rogaway, editor, *Advances in Cryptology - CRYPTO 2011 - 31st Annual Cryptology Conference, Santa Barbara, CA, USA, August 14-18, 2011. Proceedings*, volume 6841 of *Lecture Notes in Computer Science*, pages 743–760. Springer, 2011.
8. D. J. Bernstein, T. Lange, and C. Peters. Wild McEliece incognito. In Yang [27], pages 244–254.
9. W. Bosma, J. J. Cannon, and C. Playoust. The Magma algebra system I: The user language. *Journal of Symbolic Computation*, 24(3-4):235–265, 1997.
10. A. Canteaut and F. Chabaud. A new algorithm for finding minimum-weight words in a linear code: Application to McEliece's cryptosystem and to narrow-sense BCH codes of length 511. *IEEE Transactions on Information Theory*, 44(1):367–378, 1998.
11. A. Couvreur, A. Otmani, and J. Tillich. Polynomial time attack on wild McEliece over quadratic extensions. In P. Q. Nguyen and E. Oswald, editors, *Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings*, volume 8441 of *Lecture Notes in Computer Science*, pages 17–39. Springer, 2014.
12. J.-C. Faugère. A new efficient algorithm for computing gröbner bases (F4). *Journal of Pure and Applied Algebra*, 139(1-3):61–88, 1999.
13. J.-C. Faugère, A. Otmani, L. Perret, F. de Portzamparc, and J.-P. Tillich. Structural cryptanalysis of McEliece schemes with compact keys. *IACR Cryptology ePrint Archive*, 2014:210, 2014.
14. J.-C. Faugère, A. Otmani, L. Perret, and J.-P. Tillich. Algebraic cryptanalysis of Mceliece variants with compact keys. In H. Gilbert, editor, *EUROCRYPT*, volume 6110 of *Lecture Notes in Computer Science*, pages 279–298. Springer, 2010.
15. J.-C. Faugère, A. Otmani, L. Perret, and J.-P. Tillich. Algebraic Cryptanalysis of McEliece variants with compact keys – toward a complexity analysis. In *SCC '10: Proceedings of the 2nd International Conference on Symbolic Computation and Cryptography*, pages 45–55, RHUL, June 2010.
16. S. Heyse. Implementation of McEliece based on quasi-dyadic Goppa codes for embedded devices. In Yang [27], pages 143–162.
17. J. S. Leon. A probabilistic algorithm for computing minimum weights of large error-correcting codes. *IEEE Transactions on Information Theory*, 34(5):1354–1359, 1988.

18. P. Loidreau and N. Sendrier. Weak keys in the McEliece public-key cryptosystem. *IEEE Transactions on Information Theory*, 47(3):1207–1211, 2001.
19. A. May, A. Meurer, and E. Thomae. Decoding random linear codes in $\widetilde{O}(2^{0.054n})$. In D. H. Lee and X. Wang, editors, *Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings*, volume 7073 of *Lecture Notes in Computer Science*, pages 107–124. Springer, 2011.
20. R. J. McEliece. *A Public-Key System Based on Algebraic Coding Theory*, pages 114–116. Jet Propulsion Lab, 1978. DSN Progress Report 44.
21. R. Misoczki and P. S. L. M. Barreto. Compact McEliece keys from Goppa codes. In M. J. Jacobson, V. Rijmen, and R. Safavi-Naini, editors, *Selected Areas in Cryptography, 16th Annual International Workshop, SAC 2009, Calgary, Alberta, Canada, August 13-14, 2009, Revised Selected Papers*, volume 5867 of *Lecture Notes in Computer Science*, pages 376–392. Springer, 2009.
22. H. Niederreiter. Knapsack-type cryptosystems and algebraic coding theory. *Problems Control Inform. Theory*, 15(2):159–166, 1986.
23. E. Persichetti. Compact McEliece keys based on quasi-dyadic srivastava codes. *J. Mathematical Cryptology*, 6(2):149–169, 2012.
24. C. Peters. Information-set decoding for linear codes over $\mathbb{F}_q$. In N. Sendrier, editor, *Post-Quantum Cryptography, Third International Workshop, PQCrypto 2010, Darmstadt, Germany, May 25-28, 2010. Proceedings*, volume 6061 of *Lecture Notes in Computer Science*, pages 81–94. Springer, 2010.
25. N. Sendrier. Finding the permutation between equivalent linear codes: The support splitting algorithm. *IEEE Transactions on Information Theory*, 46(4):1193–1203, 2000.
26. V. Sidelnikov and S. Shestakov. On the insecurity of cryptosystems based on generalized Reed-Solomon codes. *Discrete Mathematics and Applications*, 1(4):439–444, 1992.
27. B.-Y. Yang, editor. *Post-Quantum Cryptography - 4th International Workshop, PQCrypto 2011, Taipei, Taiwan, November 29 - December 2, 2011. Proceedings*, volume 7071 of *Lecture Notes in Computer Science*. Springer, 2011.

A Appendix

A.1 A Technical Lemma

We prove a technical lemma which is useful for the proofs of Sections 3 and 4.

Lemma 1. *Let $q = p^s$ (p prime and $s > 0$), and $Q = \gamma_t z^t + \cdots + \gamma_0 \in \mathbb{F}_{q^m}[z]$ be a polynomial of degree t. For all j, it holds that:*

$$\begin{aligned} Q(z)^{p^j} &= \gamma_t^{p^j}(z^t)^{p^j} + \cdots + \gamma_0^{p^j} \\ &= \gamma_t^{p^j}(z^{p^j})^t + \cdots + \gamma_0^{p^j} \\ &= F_{(j)}(Q)(z^{p^j}). \end{aligned}$$

where $F_{(j)}(Q) = \gamma_t^{p^j} z^t + \cdots + \gamma_0^{p^j}$ is the polynomial of same degree as Q obtained by raising all the coefficients to the p^j-power.

A.2 Proof of Lemma 3

We want to prove that, under the conditions of Lemma 3 (that is $0 \leqslant u_y \leqslant q, 0 \leqslant u_x \leqslant u_y t - 1, 0 \leqslant u \leqslant \deg(f) - 1, b \in \{0,1\}$ and $(b, u_y) \neq (0,0)$), it holds that

$$\left\{ \sum_{j=0}^{n-1} g_{i,j} \left(w_i x_i^u\right)^b y_j^{u_y} x_j^{u_x} = 0 \mid 0 \leqslant i \leqslant k-1 \right\}.$$

Proof. The crucial remark is that, for any $\mathbf{c} \in \mathbb{F}_q^n$, $\sum_{i=0}^{n-1} \frac{c_i}{z - x_i} \equiv 0 \bmod f(z) \Gamma^q(z)$ implies $\sum_{i=0}^{n-1} \frac{c_i}{z-x_i} \equiv 0 \bmod f^b(z) \Gamma^{u_y}(z)$ for all $0 \leqslant u_y \leqslant q$ and $0 \leqslant b \leqslant 1$ (and $(u_y, b) \neq (0,0)$). In other words, for those u_y, b , it holds that

$$\mathscr{C}_{pub} \subseteq \mathscr{G}_q(\mathbf{x}, f^b(z)\Gamma^{u_y}(z)).$$

As $\mathscr{G}_q(\mathbf{x}, f(z)^b \Gamma^{u_y}(z))$ has parity check matrix $\mathbf{V}_{d_{tot}}(\mathbf{x}, \mathbf{w}^b \mathbf{y}^{u_y})$ (with $d_{tot} = b \deg(f) + u_y t$), the matrix products $V_{d_{tot}}(\mathbf{x}, \mathbf{w}^b \mathbf{y}^{u_y}) \times \mathbf{G}_{pub}^{\mathrm{T}} = \mathbf{0}_{d_{tot} \times k}$ yield all the relations of the lemma. □

A.3 Proof of Theorem 2

Proof. We give the multinomial development of the $z_j^u = \left(\sum_{\ell \in \mathcal{L}_a} y_j^\ell Q_\ell(x_j^\ell)\right)^u$ under the form $z_j^u = \sum_{u_x, u_y} \alpha_{u_x,u_y} y_j^{u_y} x_j^{u_x}$ and show that u_x, u_y satisfy the conditions of Lemma 3. This is done separately for $u \in \mathcal{P}_1$ and $u \in \mathcal{P}_2$.

Case $u \in \mathcal{P}_1$. We pick $u \in \{1, 2, \ldots, p^a - 1\}$ and use the multinomial formula to expand $\left(\sum_{\ell \in \mathcal{L}_a} y_j^\ell Q_\ell(x_j^\ell)\right)^u$. Namely, with $L_a = \#\mathcal{L}_a$, we have:

$$\left(\sum_{\ell \in \mathcal{L}_a} y_j^\ell Q_\ell(x_j^\ell)\right)^u = \sum_{\substack{0 \leqslant u_1, \ldots, u_L \leqslant u \\ u_1 + \ldots + u_L = u}} \left(\binom{u}{u_1, \ldots, u_L} y_j^{\left(\sum_{\ell \in \mathcal{L}_a} \ell u_\ell\right)} \prod_{\ell \in \mathcal{L}_a} Q_\ell(x_j^\ell)^{u_\ell} \right).$$

Let's look at each term $y_j^{u_y} x_j^{u_x}$ in the sum. For $u_1, \ldots, u_L$ non-negative integers with $u_1 + \ldots + u_L = u$, it holds that $u_y = \sum_{\ell \in \mathcal{L}_a} \ell u_\ell \leqslant \max(\mathcal{L}_a) \sum_{\ell \in \mathcal{L}_a} u_\ell \leqslant p^{s-a} u \leqslant p^s$. For each $y_j^{u_y}$, several terms $y_j^{u_y} x_j^{u_x}$ appear after expanding $\prod_{\ell \in \mathcal{L}_a} Q_\ell(x_j^\ell)^{u_\ell}$. In $Q_\ell(x_j^\ell)^{u_\ell}$ the maximal power u_x appearing is $\ell u_\ell (t-1)$ (as Q_ℓ has degree $t-1$). Thus, in $\prod_{\ell \in \mathcal{L}_a} Q_\ell(x_j^\ell)^{u_\ell}$, the maximal power is $(t-1) \sum_{\ell \in \mathcal{L}_a} \ell u_\ell = (t-1) u_y \leqslant t u_y - 1$.

Case $u \in \mathcal{P}_2$. We pick $b \in \{a, \ldots, s\}$. Then $z_j^{p^b} = \left(\sum_{\ell \in \mathcal{L}_a} y_j^\ell Q_\ell(x_j^\ell)\right)^{p^b} = \sum_{\ell \in \mathcal{L}_a} y_j^{\ell p^b} F_{(b)}(Q_\ell)(x_j^{\ell p^b})$ (Lemma 1). Pick $\ell \in \mathcal{L}_a$, it writes $\ell = \alpha p^c$ with $1 \leqslant \alpha < p$ and $0 \leqslant c \leqslant s - a$. Thus we have $\ell p^b = \alpha p^{c+b}$. The euclidian division of $c + b$ by s gives $c + b = ds + e$ with $0 \leqslant e < s$. The exponent ℓp^b then writes $\ell p^b = \alpha p^e p^{ds} = \alpha p^e q^d$. As $g_{i,j}^q = g_{i,j}$ it holds that $\left(\sum_{j=0}^{n-1} g_{i,j} y_j^{\alpha p^e q^d} F_{(b)}(Q_\ell)(x_j^{\alpha p^e q^d})\right)^{q^m/q^d} = \sum_{j=0}^{n-1} g_{i,j} y_j^{\alpha p^e} F_{(b-ds)}(Q_\ell)(x_j^{\alpha p^e})$. As the $F_{(b-ds)}(R_\ell)$'s have degree lower than t, all the terms of the sum are of the form $y_j^{u_y} x_j^{u_x}$ with $u_y \leqslant q$ (since $\alpha p^e < p^s$) and $u_x \leqslant u_y t - 1$. □

A.4 Proof of Proposition 6

When $p = 2$, we prove Proposition 6 without resorting to Assumption 1. We use the fact that the polynomial $\Gamma(z)$ linking $\mathbf{x}$ and $\mathbf{y}^{-1}$ is irreducible in the construction proposed in [6,8]. For $p = 2$, $\mathcal{L}_a$ is reduced to powers of 2, namely $\mathcal{L}_a = \{p^u\}_{0 \leqslant u \leqslant s-a}$. So the proof consists in showing that the intersection

$$\mathcal{I} = \left(\sum_{u=0}^{s-a} \mathrm{GRS}_t(\mathbf{x}^{p^u}, \mathbf{y}^{p^u}) \right) \cap \left(\sum_{u=s-a}^{2(s-a)} \mathrm{GRS}_t(\mathbf{x}^{p^u}, \mathbf{y}^{p^u}) \right)$$

is reduced to $\mathrm{GRS}_t(\mathbf{x}^{p^{s-a}}, \mathbf{y}^{p^{s-a}})$.

Proof. We pick $\mathbf{v} \in \mathcal{I}$. There exist polynomials $R_{p^u}, Q_{p^{s-a+u}} \in \mathbb{F}_{q^m}[z]$ (with $0 \leqslant u \leqslant s-a$) of degree lower than t such that

$$v_i = \sum_{u=0}^{s-a} y_i^{p^u} R_{p^u}(x_i^{p^u}) = \sum_{u=0}^{s-a} y_i^{p^{s-a+u}\ell} Q_{p^{s-a+u}}(x_i^{p^{s-a+u}\ell})$$

for all $0 \leqslant i \leqslant n-1$. As $y_i = \Gamma(x_i)^{-1}$, we obtain polynomial relations in the x_i's by multiplying by $\Gamma(x_i)^{p^{2(s-a)}}$. This yields n relations,

$$\sum_{u=0}^{s-a} \Gamma(x_i)^{(p^{2(s-a)-u})} R_{p^u}(x_i^{p^u}) = \sum_{u=0}^{s-a} \Gamma(x_i)^{p^{2(s-a)-(s-a+u)}} Q_{p^{s-a+u}}(x_i^{p^{s-a+u}}).$$

We suppose here that the degree of this polynomial relation is lower than n, that is $(t-1)p^{2(s-a)} < n$, so that we can deduce the polynomial equality:

$$\sum_{u=0}^{s-a} \Gamma(z)^{(p^{2(s-a)-u})} R_{p^u}(z^{p^u}) = \sum_{u=0}^{s-a} \Gamma(z)^{p^{2(s-a)-(s-a+u)}} Q_{p^{s-a+u}}(z^{p^{s-a+u}}) \qquad (7)$$

Modulo $\Gamma(z)$ all polynomials vanish but one, this yields $Q_{p^{2(s-a)}}(z^{p^{2(s-a)}}) \equiv 0 \mod \Gamma(z)$. Thanks to Lemma 1, we have $\Gamma(z)$ divides $Q_{p^{2(s-a)}}(z^{p^{2(s-a)}}) = \left(F_{(u)}(Q_{p^{2(s-a)}})(z)\right)^{p^{2(s-a)}}$ (for $u = ms - 2(s-a)$). As $\Gamma(z)$ is irreducible, this entails that $\Gamma(z)$ divides $F_{(u)}(Q_{p^{2(s-a)}})(z)$, but $F_{(u)}(Q_{p^{2(s-a)}})(z)$ has same degree as $Q_{p^{2(s-a)}}(z)$, which has degree lower than t (notations as in the proof of Theorem 2). Hence we deduce that $F_{(u)}(Q_{p^{2(s-a)}})(z) = 0$ and also its Fröbenius $Q_{p^{2(s-a)}} = 0$. Then, we look at the new relation of type (7) and start over with the polynomial $Q_{p^{2(s-a)-1}}(z^{p^{2(s-a)-1}})$. The proof that $Q_{p^{2(s-a)-1}} = 0$ is identical. One after the other, we prove that all the polynomials $R_{p^u}, Q_{p^{s-a+u}}$ are zero except the matching polynomials $R_{p^{s-a}}$ and $Q_{p^{s-a}}$ which are equal, so that $\mathbf{z} \in \mathrm{GRS}_t(\mathbf{x}^{p^{s-a}}, \mathbf{y}^{p^{s-a}})$. The problem when $p \neq 2$ is that the set $\mathcal{L}_a$ contains exponents which are not a pure power of p. □