An attack on the CFS scheme and on TII McEliece challenges

Magali Bardet[1], Axel Lemoine[2,3], and Jean-Pierre Tillich[2]

[1] Université de Rouen Normandie, LITIS UR 4108
[2] Inria, France
[3] DGA, France

Abstract. It has been a very long standing open question whether the CFS signature scheme [CFS01] whose security is basically that of a McEliece scheme based on very high rate binary Goppa codes could be attacked or not. There was a first cryptanalytic result [FGO+11] consisting in finding a distinguisher for the binary Goppa codes used in this scheme showing that these codes can be distinguished in polynomial time from a random binary linear code. However despite numerous cryptanalytic attempts [COT14, BMT24, CMT23a, Mor25, Ran25] and even if the original distinguisher has been significantly improved, no attack on the McEliece scheme based on binary Goppa codes has been found so far except for very peculiar Goppa codes of degree 2 [Mor25]. We show here that the Pfaffian modeling used in the distinguishing attack of [CMT23a] can actually be used together with a shortening trick and looking for squares in the corresponding ideal to find a polynomial attack on the CFS scheme based on very high rate binary Goppa codes.This breaks this 25 years old signature scheme. We demonstrate the effectiveness of this approach by recovering the key of TII McEliece challenges with a claimed key security of up to 210 bits.

1 Introduction

1.1 The McEliece scheme

The McEliece encryption scheme [McE78], which is only a few months younger than RSA [RSA78], is the oldest public key cryptosystem that is still resistant to a classical or quantum computer. Its underlying security assumption is radically different from RSA. It relies partly on decoding a generic linear code which is known to be NP-hard [BMvT78]. The public key is a generic description of a linear code $\mathscr{C}$ by the means of a generator matrix $\boldsymbol{G}$. In its simplest form, encryption is basically performed as

$$\mathsf{Enc} : \boldsymbol{m} \longmapsto \boldsymbol{m}\boldsymbol{G} + \boldsymbol{e},$$

where $\boldsymbol{e}$ is a random error vector whose Hamming weight is equal to the error correcting capability of $\mathscr{C}$. An efficient decoding algorithm is then used to remove the error and recover the original message $\boldsymbol{m}$. This algorithm must be kept secret

and plays the role of the private key. The public description of the code should obviously not leak any information about the secret decoding algorithm. On the other hand, the code cannot be random, as this would prevent the existence of an efficient decoding algorithm. In the original proposal, McEliece suggested the use of binary Goppa codes. They indeed benefit from a good error-correcting capability, and their weight distribution is close to that of random codes.

This scheme has withstood tremendous cryptanalytic efforts, be they (i) combinatorial key recovery attacks [Gib91, Sen00, Pan25], (ii) algebraic key recovery attacks [FOPT10, COT14, FPdP14, BMT24, CMT23b, Mor25, Lem25], (iii) message recovery attacks [LB88, Ste88, Dum89, CC98, MMT11, BJMM12, MO15, BM17, CDMT22, Ess22, EZ23, CDMT24, DEEK24] that hardly decreased the exponent of these attacks which are exponential with respect to the decoding capacity of the underlying Goppa code used in the scheme, (iv) clever implementations of message recovery attacks that attacked successfully the original McEliece parameters [BLP08] resulting in a speed-up of several orders of magnitude of the best attacks known at that time but without changing the exponent of the best message recovery attacks which can be easily thwarted by a slight increase of the parameters, quantum key recovery attacks [DMR11] or quantum message attacks [Ber10, KT17, Tra24]. This very strong resistance against attacks was a very convincing argument in the NIST post-quantum cryptography competition for the variation of this public key cryptosystem intended to be IND-CCA secure and an associated key exchange protocol `Classic McEliece` [ABC+22] which went into the final round of this competition. Note that, although it was ultimately not chosen by NIST for standardization, it is currently undergoing a standardization process by the ISO.

1.2 An overview of algebraic cryptanalysis of related schemes

The Goppa distinguishing problem. The scheme is provably secure under the assumption that the family of codes is indistinguishable from that of random linear codes with identical parameters [CFS01]. The family of binary Goppa codes provides indeed a good candidate, for they share many properties with random codes, such as their weight distribution [LL97, Fin03]. It was for a long time believed that this problem was intractable before [FGO+11] showed that it could be done in polynomial time for the very high rate Goppa codes used in the CFS signature scheme [CFS01]. This did not provide an actual attack on the scheme but undermined its security proof.

The interesting feature of this line of research is that it is apparently a somewhat simpler problem than the formidable problem of breaking the McEliece scheme itself. Moreover, it has proved fruitful for making progress on understanding the features of the Goppa codes which could be used to eventually mount an attack. While the distinguisher of [FGO+11] was only able to distinguish Goppa codes of rate $1 - \widetilde{\mathcal{O}}\left(\frac{1}{\sqrt{n}}\right)$ where n is the codelength, some recent developments [CMT23b, Ran25, Wie25] extended the regime where distinguishing is doable in polynomial time. These new distinguishers even apply in the constant rate

regime, but then lose their polynomial time complexity. It is worthwhile to note that they are able to distinguish the Goppa codes used in the McEliece scheme, sometimes at a cost which is smaller than the state-of-the-art key attacks which are based on the support splitting algorithm [Sen00]. In particular, key distinguishing in subexponential complexity with respect to the decoding capacity of the underlying Goppa code was achieved for the first time in the breakthrough paper [Ran25]. This is remarkable since the complexities of the best message attacks are exponential in this quantity. Note however that this is an asymptotic result, and it does not directly threaten the security of the NIST submission `Classic McEliece` for which the actual complexity of the [Ran25] distinguisher is significantly higher than the security target 2^{λ} which corresponds in this case to the complexity of the best message attacks.

Alternant codes and Goppa codes. Interestingly enough these distinguishers also distinguish a slightly more general family of codes than Goppa codes, namely alternant codes. Indeed, the latter family really carries the essential polynomial structure which is the distinctive feature of Goppa codes that makes it amenable to this family of distinguishers. Let us recall the definition of such codes. Basically alternant codes are a way to keep some of the nice decoding properties of the family of Reed-Solomon codes while being defined over a fixed alphabet. Indeed, such codes are defined as subfield subcodes of (generalized) Reed-Solomon codes.

Definition 1.1 (Generalized Reed-Solomon code). *Let $\boldsymbol{x} = (x_1, \ldots, x_n) \in \mathbb{F}^n$ be a vector of pairwise-distinct elements, $\boldsymbol{y} = (y_i)_{1 \leq i \leq n}$ a vector of nonzero elements and $r < n$ be an integer. The Generalized Reed-Solomon code (GRS) of degree r, support $\boldsymbol{x}$ and multiplier $\boldsymbol{y}$ is defined as*

$$\mathbf{GRS}_r(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\text{def}}{=} \{(y_i f(x_i))_{1 \leq i \leq n} \mid f \in \mathbb{F}[X]_{<r}\}.$$

Alternant codes are defined as subfield subcodes of GRS codes defined over an extension field $\mathbb{F}_{q^m}$, *i.e.* if $\mathscr{C}$ is a GRS code defined over $\mathbb{F}_{q^m}$ then the associated alternant code $\mathscr{A}$ over $\mathbb{F}_q$ is nothing but

$$\mathscr{A} = \mathscr{C} \cap \mathbb{F}_q^n,$$

where n is the length of $\mathscr{C}$ and $\mathscr{A}$. Obviously any decoding algorithm for the GRS code can be used to decode the alternant code. It is a little bit more convenient to define such codes through the dual code by using the fact that the dual of a GRS code is itself a GRS code (see Proposition 2.2).

Definition 1.2 (Alternant code). *Let $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ be a vector whose entries are all distinct, $\boldsymbol{y} \in (\mathbb{F}_{q^m}^{\times})^n$ and let $r < n$ be an integer. The alternant code over $\mathbb{F}_q$ of support $\boldsymbol{x}$, multiplier $\boldsymbol{y}$, degree r and extension degree m is defined by*

$$\mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y}) \stackrel{\text{def}}{=} \left(\mathbf{GRS}_r(\boldsymbol{x}, \boldsymbol{y})^{\perp}\right) \cap \mathbb{F}_q^n. \tag{1}$$

The dimension of the alternant code satisfies $\dim_{\mathbb{F}_q} \mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y}) \geq n - rm$. For generic alternant codes, this bound holds actually with equality. Now, Goppa codes are nothing but alternant codes with an algebraic relation between the support and the multiplier.

Definition 1.3 (Goppa code). *Let $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$ be a vector whose entries are all distinct. Let $r < n$ be an integer and $\Gamma \in \mathbb{F}_{q^m}[X]$ be some monic polynomial such that $\Gamma(x_i) \neq 0,\ i = 1, \ldots, n$. The Goppa code over $\mathbb{F}_q$ of support $\boldsymbol{x}$ and Goppa polynomial Γ is defined as*

$$\mathscr{G}(\boldsymbol{x}, \Gamma) \stackrel{\text{def}}{=} \mathscr{A}_r\left(\boldsymbol{x}, \frac{1}{\Gamma(\boldsymbol{x})}\right), \tag{2}$$

where $\dfrac{1}{\Gamma(\boldsymbol{x})}$ *is the vector* $\left(\dfrac{1}{\Gamma(x_i)}\right)_{1 \leq i \leq n}$.

Interestingly enough, it turns out that the slightly more structured Goppa codes behave a little bit better for the distinguisher of [FGO+13].

Attacks on variants of the McEliece scheme. The original McEliece scheme is based on binary Goppa codes. Over the years, many variants were proposed in order to decrease the size of the public key, such as quasi-cyclic alternant or Goppa codes [BCGO09, CBB+17] or quasi-dyadic Goppa codes [MB09, BLM11, BBB+17]. Another way was to use Goppa codes defined over a larger alphabet and with very small extension degree and better decoding capacities, resulting in smaller codes and smaller public keysizes [BLP10, BLP11].

Thanks to the algebraic polynomial structure of alternant codes, the problem consisting in retrieving a support and a multiplier from the mere knowledge of the public key can be modelized by an algebraic system [FOPT10] which can be solved with Gröbner basis techniques. This approach was successful in the quasi cyclic/dyadic case [BCGO09, MB09], as this structure allowed to reduce drastically the number of unknowns of the algebraic system when compared to the original McEliece cryptosystem. Another attack [BC18] also exploited in a crucial way the quasi-dyadic structure of the `DAGS` NIST submission [BBB+17]. A variant of this algebraic modeling was introduced in [FPdP14] to attack certain parameters of the variant of the McEliece cryptosystem [BLP10, BLP11] based on wild Goppa codes and wild Goppa codes incognito. The McEliece cryptosystem based on plain binary Goppa codes seems immune to both [FOPT10] and [FPdP14] approaches. The first one because the degree and the number of variables of the resulting system are most certainly too big to make such an approach likely to succeed and the second one because this modeling does not apply to binary Goppa codes.

The square code distinguisher and attack. The polynomial structure of alternant codes and Goppa codes makes them susceptible to square attacks. Basically this comes from considering the algebra on $\mathbb{F}_q^n$ based on the following seemingly uninteresting component-wise product of vectors

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

with $\boldsymbol{a} = (a_i)_{1\le i\le n}$ and $\boldsymbol{b} = (b_i)_{1\le i\le n}$. There is an associated product of codes which is defined as

Definition 1.4. *The component-wise product of codes $\mathscr{C}, \mathscr{D}$ over $\mathbb{F}_q$ with the same length n is defined as*

$$\mathscr{C} \star \mathscr{D} \stackrel{\text{def}}{=} \langle\, \boldsymbol{c} \star \boldsymbol{d} \mid \boldsymbol{c} \in \mathscr{C}, \boldsymbol{d} \in \mathscr{D} \,\rangle_{\mathbb{F}_q}.$$

If $\mathscr{C} = \mathscr{D}$, we call $\mathscr{C}^{\star 2} \stackrel{\text{def}}{=} \mathscr{C} \star \mathscr{C}$ the square code of $\mathscr{C}$.

It can be verified [CCMZ15] that for a random code of length n we typically have $\dim \mathscr{C}^{\star 2} = \min\left\{n, \binom{\dim \mathscr{C}+1}{2}\right\}$, meaning that the square code of a random code quickly fills up the whole space when the dimension becomes bigger than $\sqrt{2n}$. On the other hand it is easy to see that the polynomial structure of GRS codes gives a much smaller dimension. We namely have

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

which implies that

$$\dim\left(\mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y})^{\star 2}\right) = \min\left(2 \cdot \dim \mathbf{GRS}_k(\boldsymbol{x}, \boldsymbol{y}) - 1, n\right).$$

This property was used for the first time in [Wie10] for cryptanalyzing the Berger-Loidreau variant [BL04] of the McEliece cryptosystem replacing the binary Goppa code by a subcode of a GRS code. It was also used in [CGG+14] to mount an attack on McEliece or Niederreiter schemes based on GRS codes [Nie86] different from the original cryptanalysis [SS92]. This property that the square code of a GRS code does not fill the whole space as soon as the rate is below $\frac{1}{2}$ can also be used to distinguish such algebraic codes from random codes. This property survives for the dual of very high rate alternant or Goppa codes. It was shown in [MP12] that this square code approach for alternant and Goppa codes is equivalent to the original distinguisher [FGO+11] (see also [MT22]).

It also turned out that suitable shortened (see Definition 2.1) wild Goppa codes also have an abnormally low dimension when the degree of extension is only 2 [COT14]. In this case, working directly on the Goppa code itself did not only provide a distinguisher, it also provided an attack [COT14]. The situation was much more complicated for the distinguisher of [FGO+11] where it took more than ten years to transform the distinguisher of [FGO+11] into an attack [BMT24]. This attack works in the case of generic alternant codes that are "square-distinguishable" over $\mathbb{F}_2$ and $\mathbb{F}_3$. Amazingly enough, this attack completely fails in the case of Goppa codes which forms a more structured subfamily of alternant codes.

Matrices of low rank in the code of quadratic relationships (aka the quadratic hull). In characteristic 2, a generalization of the distinguisher of [FGO+11] has been proposed in [CMT23b]. It expands significantly the range of alternant and Goppa codes that can be distinguished by it. There are four ingredients. The first one considers the extension of the dual code over the extension field $\mathbb{F}_{q^m}$ for an alternant/Goppa code defined over $\mathbb{F}_q$. Such code is defined by

Definition 1.5 (Extension of a code over a field extension). *Let $\mathscr{C}$ be a linear code over $\mathbb{F}_q$. We denote by $\mathscr{C}_{\mathbb{F}_{q^m}}$ the $\mathbb{F}_{q^m}$-linear span of $\mathscr{C}$ in $\mathbb{F}_{q^m}^n$.*

The point is that the extension over $\mathbb{F}_{q^m}$ of the dual of an alternant code of extension degree m over $\mathbb{F}_q$ contains the underlying GRS code, as

$$(\mathscr{A}_r(\boldsymbol{x},\boldsymbol{y})^\perp)_{\mathbb{F}_{q^m}} = \sum_{j=0}^{m-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{q^j}, \boldsymbol{y}^{q^j}\right), \tag{3}$$

(see for instance [BMT24, prop. 14]). The second ingredient then notices that we have triples of codewords $\boldsymbol{c}_1, \boldsymbol{c}_2, \boldsymbol{c}_3$ in this extended code that satisfy

$$\boldsymbol{c}_1 \star \boldsymbol{c}_2 = \boldsymbol{c}_3^2. \tag{4}$$

An example of such a triple is obviously $\boldsymbol{c}_1 = \boldsymbol{y},\ \boldsymbol{c}_2 = \boldsymbol{y} \star \boldsymbol{x}^2,\ \boldsymbol{c}_3 = \boldsymbol{y} \star \boldsymbol{x}$. One could argue that these triples lie in the private basis of the code. Here comes the third ingredient. From an arbitrary basis $\mathcal{B} = (\boldsymbol{b}_1, \cdots, \boldsymbol{b}_k)$ of the extension $\mathscr{C}$ over $\mathbb{F}_{q^m}$ of the dual of the alternant or Goppa code we want to distinguish, one computes the vector space formed by all vectors $\boldsymbol{c} = (c_{ij})_{\substack{i,j\in[\![1,k]\!]\\ i\le j}} \in \mathbb{F}_{q^m}^{k(k+1)/2}$ such that $\sum_{i\le j} c_{ij}\boldsymbol{b}_i \star \boldsymbol{b}_j = 0$. We call this space the *code of quadratic relations of $\mathscr{C}$ with respect to $\mathcal{B}$* and denote it with $\mathscr{C}_{\mathrm{rel}}(\mathcal{B})$. In other words this is the linear space $\mathscr{Q}$ of quadratic relations involving $(\boldsymbol{c}_1, \cdots, \boldsymbol{c}_k)$. It is also called the quadratic hull of $\mathscr{C}$. Elements $\boldsymbol{c}$ of this space can be viewed as a quadratic form

$$Q_{\boldsymbol{c}}(\boldsymbol{x}) = \sum_{i\le j} c_{ij} x_i x_j,$$

or as symmetric matrices by taking the associated polar form $\boldsymbol{M}_{\boldsymbol{c}}$ defined by

$$\boldsymbol{x}\boldsymbol{M}_{\boldsymbol{c}}\boldsymbol{y}^\intercal = Q_{\boldsymbol{c}}(\boldsymbol{x}+\boldsymbol{y}) - Q_{\boldsymbol{c}}(\boldsymbol{x}) - Q_{\boldsymbol{c}}(\boldsymbol{y}),$$

where $\boldsymbol{x} \stackrel{\text{def}}{=} (x_1, \cdots, x_k)$ and $\boldsymbol{y} \stackrel{\text{def}}{=} (y_1, \cdots, y_k)$. We denote this space of polar forms as the matrix code of quadratic relations associated to $\mathscr{C}$ and the basis $\mathcal{B}$:

Definition 1.6 (Matrix code of relations). *Let $\mathscr{C}$ be an $[n,k]$ linear code over $\mathbb{F}$ with basis $\mathcal{B} = \{\boldsymbol{b}_1, \ldots, \boldsymbol{b}_k\}$. The **matrix code of quadratic relations with respect to** $\mathcal{B}$ is*

$$\mathscr{C}_{\mathrm{mat}}(\mathcal{B}) \stackrel{\text{def}}{=} \{\boldsymbol{M}_{\boldsymbol{c}} = (m_{i,j})_{\substack{1\le i\le k\\ 1\le j\le k}} \mid \boldsymbol{c} = (c_{i,j})_{1\le i\le j\le k} \in \mathscr{C}_{\mathrm{rel}}(\mathcal{B})\} \subseteq \mathbf{Sym}(k,\mathbb{F}),$$

where $\boldsymbol{M}_{\boldsymbol{c}}$ is defined as $m_{i,j} \stackrel{\text{def}}{=} m_{j,i} \stackrel{\text{def}}{=} c_{i,j}, i \neq j$ and $m_{i,i} \stackrel{\text{def}}{=} 2c_{i,i}, 1 \le i \le k$.

Assume now we have chosen a basis $\mathcal{A}$ of $\mathscr{C}$ where the first three elements are $\boldsymbol{c}_1 = \boldsymbol{y},\ \boldsymbol{c}_2 = \boldsymbol{y} \star \boldsymbol{x}^2,\ \boldsymbol{c}_3 = \boldsymbol{y} \star \boldsymbol{x}$. In characteristic 2, it is readily seen that the matrix of the polar form corresponding to the quadratic relation (4) is 0

everywhere except in the first 3×3 block where it is equal to $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$, giving a matrix of rank 2. In an arbitrary basis $\mathcal{B} = (\boldsymbol{b}_1, \cdots, \boldsymbol{b}_k)$ the polar form of the quadratic relation (4) would still be a matrix of rank 2 because if $\mathcal{A}$ and $\mathcal{B}$ are two different bases of the same code, there exists (see [CMT23b, Prop. 4]) an invertible $\boldsymbol{P} \in \mathbb{F}_{q^m}^{k \times k}$ such that

$$\mathscr{C}_{\mathrm{mat}}(\mathcal{A}) = \boldsymbol{P}^{\intercal} \mathscr{C}_{\mathrm{mat}}(\mathcal{B}) \boldsymbol{P}. \tag{5}$$

This is the third ingredient, namely that the matrix code of quadratic relations contains rank 2 matrices, regardless of the basis $\mathcal{B}$.

The fourth ingredient consists in finding such a matrix of rank 2 in the matrix code of quadratic relations. This is a MinRank problem associated to the matrix code $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$. The key fact here is that this is a space of antisymmetric matrices when the field characteristic is 2. We use an algebraic approach by

(1) expressing an element $\boldsymbol{M}$ of $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ as a $k \times k$ matrix with entries that are linear forms in $(z_1, \ldots, z_K)$ where the z_i's are such that

$$\boldsymbol{M} = z_1 \boldsymbol{M}_1 + \cdots + z_K \boldsymbol{M}_K$$

with $\boldsymbol{M}_1, \cdots, \boldsymbol{M}_K$ being a basis of antisymmetric matrices of $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$,

(2) writing a quadratic system in $z_1, \cdots, z_K$ expressing the fact that every 4×4 principal minor of such an element of rank 2 in $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ should be zero. The determinant of this antisymmetric submatrix of size 4 is indeed the square of what is called a Pfaffian and is therefore a quadratic form.

The distinguisher consists in observing that solving this algebraic system by Gröbner basis techniques differs from solving a quadratic system with the same number of unknowns and equations [CMT23b, §5]. It basically amounts to the distinguisher of [FGO+11] in the very high rate alternant or Goppa code regime, stays subexponential in the error correction code capacity when the dimension s of the dual satisfies $s = \mathcal{O}\left(n^{1-\varepsilon}\right)$ for some constant $\varepsilon > 0$.

TII challenges. TII published on its website[4] in 2023 a whole set of challenges aiming at better understanding the security of the McEliece cryptosystem and fostering further research on the topic. There were two kinds of contributions to the topic of the key attack challenges:

– An algebraic attack [Mor25] using the Pfaffian modeling and distinguisher of [CMT23b]. It consisted in solving with Gröbner basis techniques the resulting algebraic system by specializing a suitable number of variables and finding for a given specialization a rank 2 matrix in the matrix code of quadratic relations. By finding two matrices of rank 2, using two different specializations, the support and Goppa polynomial of the Goppa code could

[4] https://crowdchallenge.tii.ae/mceliece-challenges

be recovered. This attack only applies to the very specific case $r = 2$ which concerns only some very small parameters with claimed security ranging between 22 and 69 bits. All the parameter sets corresponding to $r = 2$ were broken, the attack taking only a few seconds (see [Mor25, table I]) except for one. This showed that the Pfaffian modeling of [CMT23b] is not only restricted to give a distinguisher but can be transformed into an actual attack on the McEliece scheme when $r = 2$.

– A clever and highly optimized implementation [Pan25] of the support splitting attack of [Sen00] which first finds by brute force the Goppa polynomial and then solves the code equivalence problem between the public code and the Goppa code related to a putative Goppa polynomial. It was able to break a challenge of up to 91 bits of claimed security in about 1470 core days of computation. This time the attack was not restricted to $r = 2$. For instance, the attacked instance corresponded to $r = 6$, $m = 8$, $n = 253$.

1.3 Contributions

Our main contribution is to show that very high rate binary Goppa codes can indeed be attacked successfully. The attack works in polynomial time in the distinguisher regime of [FGO+11]. It basically breaks the CFS scheme [CFS01] in polynomial time which has withstood quarter of a century of cryptanalytic efforts and is able to attack successfully some difficult TII challenges as shown by the following table.

n	m	r	Estimated security by TII	Attack [Pan25]	Our attack	Language
253	8	5	83 bits	$2^{19.06}$s $\approx$ 6.3 d	320.180s / 4.2GB	Magma
252	8	5	89 bits	$2^{24.94}$s $\approx$ 1.02 yr	315.420s / 4.3GB	Magma
507	9	7	125 bits	N/A	34088.150s / 223GB	Magma
500	9	7	166 bits	N/A	33123.319s / 213GB	Magma
491	9	7	210 bits	N/A	15508.471s / 12GB	C

Table 1: Summary of the attacks against the TII challenges. We gave for the same parameters the complexity of the very carefully implemented support splitting algorithm given in [Pan25, Table 2] which held previous records in the TII key challenges. Our attack is a `Magma` script running on a machine equipped with an Intel Xeon Gold 6240L (Cascade Lake-SP), x86_64, 2.60GHz, 4 CPUs/node, 18 cores/CPU and 6.0TiB of RAM. We give its time and memory complexity. The C implementation makes heavily use of the `m4ri` library `https://github.com/malb/m4ri`.

The bottleneck of our approach is the echelonization of a rather large binary matrix (Step 3 of Algorithm 2), although the size of the matrix is still polynomial in r and m. We give below a pessimistic estimate of how large these matrices can get for the cryptographic parameters of CFS.

m	r	# rows	# columns
16	9	$\approx 2,980,000$	1,091,503
18	9	$\approx 4,360,000$	1,314,631
20	8	$\approx 3,540,000$	808,356
24	10	$\approx 18,150,000$	3,686,970

Table 2: Size of the matrices involved in Step 3 of Algorithm 2. Performing linear algebra over binary matrices of such size ($\mathcal{O}(\log r)$ times) is enough to recover a CFS private key. The number of rows we give comes from Heuristic 2. The number of columns is the number of monomials of degree ≤ 2 in the underlying polynomial ring. These estimations are pessimistic because the bound of Theorem 3.7 proved to not be tight in our experiments, which suggests that we might shorten even more.

Results on Tables 1 and 2 were obtained with our implementation written in Magma and C, available at `https://github.com/Crypto2026submission/CFS_TII`.

The approach. Basically we use the Pfaffian modeling of [CMT23b] for finding rank 2 matrices in the matrix code of quadratic relations and use this knowledge in order to recover the support and Goppa polynomial with several new ingredients which are essential for performing the attack.

- Instead of performing directly a hybrid attack by specifying some of the unknowns in the Pfaffian modeling as in [Mor25], we first shorten the dual code in a relatively high number of positions. A lower bound on the number of times we can shorten is given in Theorems 3.5 (alternant case) and 3.7 (Goppa case). Note that this is also the crucial ingredient used in [Ran25] to obtain a distinguisher whose complexity is subexponential in the error correction capacity of the Goppa code, thus beating asymptotically the best known attacks on the scheme which are exponential in this quantity. It turns out that the Pfaffian modeling of [CMT23b] can also be improved by shortening the code. This first step reduces significantly the number of unknowns.
- We use here the explicit Pfaffian modeling of [LMT25] which reduces the number of unknowns and specify just one unknown in the homogeneous degree 2 system.
- In both the CFS and the TII challenges that we attack, the number of equations is significantly higher than the number of monomials appearing in all of them, leading lots of redundancy. We can thus afford to generate only a subset of all possible equations.
- Then instead of performing the standard Gröbner computation, we observed that the algebraic system is highly structured and contains many equations of degree 2 that are squares, *i.e.* of the form $f_i(z_1, \cdots, z_K)^2 = 0$, where f_i is an affine equation in the unknowns $z_1, \cdots, z_K$. Due to squaring being linear in characteristic 2, we can find such equations by solving a mere linear

system. Once these equations have been obtained, we use them for reducing the unknowns in the quadratic equations even further. This step is crucial to finish the computation of the Gröbner basis at degree 2 and is a key step of the attack. In other words, we have now enough quadratic equations to finish the computation without having to consider higher degree equations. Our algorithm has therefore polynomial complexity with respect to the degree r of the Goppa code and its extension degree m.

2 Notation and Algebraic Coding Preliminaries

2.1 General notation

General notation. $[\![a, b]\!]$ indicates the closed integer interval between a and b. $\mathbb{F}_q$ denotes the finite field with q elements.

Vector and matrix notation. Vectors are indicated by lowercase bold letters such as $\boldsymbol{x}$ and matrices by uppercase bold letters such as $\boldsymbol{M}$. For a square matrix $\boldsymbol{M} = (m_{ij})_{\substack{1\le i\le n\\ 1\le j\le n}}$ and an increasing subset $I \subset [\![1, n]\!]$ of columns of $\boldsymbol{M}$, we denote by $\boldsymbol{M}[I]$ the square submatrix of $\boldsymbol{M}$ given by $\boldsymbol{M}[I] \stackrel{\text{def}}{=} (m_{ij})_{i,j\in I}$. We will also simplify the notation and just write $\boldsymbol{M}[i, j, k, \ell]$ for $\boldsymbol{M}[\{i, j, k, \ell\}]$ when $I = \{i, j, k, \ell\}$ is an increasing subset of columns of $\boldsymbol{M}$.

Given a function f acting on $\mathbb{F}_q$ and a vector $\boldsymbol{x} = (x_i)_{1\le i\le n} \in \mathbb{F}_q^n$, the expression $f(\boldsymbol{x})$ is the component-wise mapping of f on $\boldsymbol{x}$, i.e. $f(\boldsymbol{x}) = (f(x_i))_{1\le i\le n}$. We will also apply this for functions f acting on $\mathbb{F}_q^n \times \mathbb{F}_q^n$: for instance for two vectors $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathbb{F}_q^n$ and two positive integers a and b we denote by $\boldsymbol{x}^a\boldsymbol{y}^b$ the vector $(x_i^a y_i^b)_{1\le i\le n}$. We will use the same operation over matrices, but in order to avoid confusion with the matrix product, we use for a matrix $\boldsymbol{A} = (a_{i,j})_{i,j}$ the notation $\boldsymbol{A}^{(s)}$ which stands for the entries of $\boldsymbol{A}$ all raised to the power s, i.e. the entry (i, j) of $\boldsymbol{A}^{(s)}$ is equal to $a_{i,j}^s$. The scalar product between $\boldsymbol{x} = (x_i)_{1\le i\le n} \in \mathbb{F}_q^n$ and $\boldsymbol{y} = (y_i)_{1\le i\le n} \in \mathbb{F}_q^n$ is denoted by $\boldsymbol{x} \cdot \boldsymbol{y}$ and is defined by $\boldsymbol{x} \cdot \boldsymbol{y} = \sum_{i=1}^n x_i y_i$.

Ideals. Ideals are indicated by calligraphic $\mathcal{I}$. Given a sequence S of polynomials, $\mathcal{I}(S)$ refers to the polynomial ideal generated by such sequence. Given the polynomials $f_1, \ldots, f_m$, we denote by $\mathcal{I}(f_1, \ldots, f_m)$ the ideal generated by them.

Codes. A linear code $\mathscr{C}$ of length n and dimension k over $\mathbb{F}_q$ is a k dimensional subspace of $\mathbb{F}_q^n$. We refer to it as an $[n, k]$-code. Linear codes can be specified compactly by a *generator matrix*, *i.e.* a $k \times n$ matrix $\boldsymbol{G}$ over $\mathbb{F}_q$ whose rows generate the associated k-dimensional linear code $\mathscr{C}$ over $\mathbb{F}_q$ or equivalently

$$\mathscr{C} = \{\boldsymbol{u}\boldsymbol{G} : \ \boldsymbol{u} \in \mathbb{F}_q^k\}.$$

The *dual* of a code $\mathscr{C}$ is defined as $\mathscr{C}^\perp \stackrel{\text{def}}{=} \{\boldsymbol{d} \in \mathbb{F}^n \mid \forall \boldsymbol{c} \in \mathscr{C},\ \boldsymbol{c} \cdot \boldsymbol{d} = 0\}$. We also consider the *shortening* operation which reduces the dimension and length of a code

Definition 2.1 (Shortening). *Let $\mathscr{C}$ be an $[n,k]$-code and $I \subset \{1,...,n\}$. The shortening of $\mathscr{C}$ at I is the $[n-|I|, \geq k-|I|]$-linear code given by*

$$\mathbf{Sh}_I(\mathscr{C}) \stackrel{\text{def}}{=} \{(c_j)_{j\notin I} \in \mathbb{F}_q^{n-|I|} \mid \boldsymbol{c} \in \mathscr{C},\ \forall i \in I,\ c_i = 0\}.$$

2.2 Properties of GRS, alternant and Goppa Codes

Alternant and Goppa codes are subfield subcodes of GRS codes as explained in the introduction. The definition given there relies on the fact that the dual of a GRS code is itself a GRS code as shown by

Proposition 2.2 ([MS86], Theorem 4, p. 304). *Let $\mathbf{GRS}_r(\boldsymbol{x},\boldsymbol{y})$ be a GRS code of length n. We have for any $r \in [\![0,n]\!]$*

$$\mathbf{GRS}_r(\boldsymbol{x},\boldsymbol{y})^\perp = \mathbf{GRS}_{n-r}\left(\boldsymbol{x},\boldsymbol{y}^\perp\right), \tag{6}$$

where $y_i^\perp = \left(y_i \prod_{j\neq i}(x_i - x_j)\right)^{-1}$, $i = 1,\ldots,n$.

The dimension of an alternant code of length n, degree r and extension degree m satisfies obviously for any support $\boldsymbol{x} \in \mathbb{F}_{q^m}^n$, multiplier $\boldsymbol{y} \in \left(\mathbb{F}_{q^m}^\times\right)^n$,

$$\dim \mathscr{A}_r(\boldsymbol{x},\boldsymbol{y}) \geq n - r\cdot m. \tag{7}$$

Definition 2.3 (proper alternant/Goppa code). *An alternant code $\mathscr{A}_r(\boldsymbol{x},\boldsymbol{y})$ of length n, degree r and extension degree m is said to be proper if and only if*

$$\dim \mathscr{A}_r(\boldsymbol{x},\boldsymbol{y}) = n - r\cdot m.$$

It holds that generic alternant and Goppa codes are proper. In this paper we only consider proper alternant and Goppa codes. In this case, we have

$$(\mathscr{A}_r(\boldsymbol{x},\boldsymbol{y})^\perp)_{\mathbb{F}_{q^m}} = \bigoplus_{j=0}^{m-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{q^j},\boldsymbol{y}^{q^j}\right). \tag{8}$$

This follows on the spot from the fact that $(\mathscr{A}_r(\boldsymbol{x},\boldsymbol{y})^\perp)_{\mathbb{F}_{q^m}}$ is the sum of the codes in the right-hand term (see (3)), and that for a linear code over $\mathbb{F}_q$ we have $\dim_{\mathbb{F}_q}(\mathscr{C}) = \dim_{\mathbb{F}_{q^m}}\left(\mathscr{C}_{\mathbb{F}_{q^m}}\right)$. Then $\dim\left(\mathscr{A}_r(\boldsymbol{x},\boldsymbol{y})^\perp_{\mathbb{F}_{q^m}}\right) = r\cdot m$, and as $\dim\left(\mathbf{GRS}_r\left(\boldsymbol{x}^{q^j},\boldsymbol{y}^{q^j}\right)\right) = r$ we deduce that the sum is direct.

2.3 The square code distinguisher

Since our attack applies to high rate Goppa codes, it is worthwhile recalling some facts about them coming from [FGO+11]. This original distinguisher essentially identifies a regime of parameters where squares of duals of alternant and Goppa codes have lower dimensions than squares of random codes with identical parameters. This comes from

Theorem 2.4 ([MT22]). *Let $\mathscr{C} = \mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y})^\perp$ be a proper alternant code of extension degree m. Then*

$$\dim \mathscr{C}^{\star 2} \leq \binom{rm+1}{2} - \frac{m}{2}(r-1)\left((2e_{\mathscr{A}}+1)r - 2\frac{q^{e_{\mathscr{A}}+1}-1}{q-1}\right), \tag{9}$$

where

$$e_{\mathscr{A}} \stackrel{\text{def}}{=} \max\{i \in \mathbb{N} \mid r \geq q^i + 1\} = \left\lfloor \log_q(r-1) \right\rfloor. \tag{10}$$

If now $\mathscr{C} = \mathscr{G}(\boldsymbol{x}, \Gamma)^\perp$ is proper, we have

$$\dim \mathscr{C}^{\star 2} \leq \binom{rm+1}{2} - \frac{m}{2}(r-1)(r-2), \qquad \text{if } r < q-1 \tag{11}$$

$$\dim \mathscr{C}^{\star 2} \leq \binom{rm+1}{2} - \frac{m}{2}r\left((2e_{\mathscr{G}}+1)r - 2(q-1)q^{e_{\mathscr{G}}-1} - 1\right), \qquad \text{otherwise,} \tag{12}$$

where

$$e_{\mathscr{G}} \stackrel{\text{def}}{=} \min\{i \in \mathbb{N} \mid r \leq (q-1)^2 q^i\} + 1 = \left\lceil \log_q\left(\frac{r}{(q-1)^2}\right)\right\rceil + 1. \tag{13}$$

On the contrary, the square of a random $[n, rm]$-code has dimension $\min\{n, \binom{rm+1}{2}\}$ with overwhelming probability [CCMZ15]. As a consequence if the length n is greater than the right-hand side of whichever inequality of Theorem 2.4 applies to $\mathscr{C}$, then computing the square code provides a distinguisher. This is the square distinguisher of [FGO+11].

An $[n, k]$-code $\mathscr{C}$ having an unusually small square code can be seen as $\mathscr{C}$ having an unusually large quadratic hull. One can realize this is by considering the evaluation map with respect to some generator matrix $\boldsymbol{G} \in \mathbb{F}_q^{k\times n}$ of $\mathscr{C}$

$$\boldsymbol{e}_{\boldsymbol{G}} : \mathbb{F}_q[X_1, \ldots, X_k] \longrightarrow \mathbb{F}_q^n \tag{14}$$

defined as the unique homomorphism of algebras such that $X_i \mapsto \boldsymbol{c}_i$, where $(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_k)$ are the rows of $\boldsymbol{G}$. We sometimes write $f(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_k)$ for $\boldsymbol{e}_{\boldsymbol{G}}(f)$. Clearly the image of the space of homogeneous polynomials of degree 2 (aka quadratic forms) under $\boldsymbol{e}_{\boldsymbol{G}}$ is exactly $\mathscr{C}^{\star 2}$, and the quadratic hull of $\mathscr{C}$ can be seen as the kernel of $\boldsymbol{e}_{\boldsymbol{G}}$ restricted to this space. More generally, if we denote[5] by $I(\mathscr{C})$ the entire kernel of $\boldsymbol{e}_{\boldsymbol{G}}$, then one has

$$I(\mathscr{C}) = \bigoplus_{r \geq 0} I_r(\mathscr{C}),$$

where $I_r(\mathscr{C})$ stands for the kernel of $\boldsymbol{e}_{\boldsymbol{G}}$ restricted to the space of homogeneous polynomials of degree r. Throughout the paper we identify the code of quadratic

[5] There is a slight abuse of notation here, as $I(\mathscr{C})$ actually depends on the choice of generator matrix $\boldsymbol{G}$. As noticed earlier and also in [Ran25], we can omit this dependency up to a linear change of variables.

relations, or quadratic hull, with $I_2(\mathscr{C})$. Note that the rank-nullity theorem implies

$$\dim I_2(\mathscr{C}) = \binom{k+1}{2} - \dim \mathscr{C}^{\star 2},$$

hence explaining why a smaller square code is equivalent to a larger quadratic hull.

2.4 Determinantal Modeling

There are several ways to explain why duals of alternant and Goppa codes have a rather large quadratic hull. One of them, introduced in [Ran25] and reused in this work, is through a certain *determinantal modeling*. The point of this approach is that it can be shown that $I_2(\mathscr{C})$ contains all 2×2 minors of a certain $2 \times f$ matrix $\mathbf{\Phi}$ whose entries are linear forms in the X_i's, and where f is an integer that will be detailed thereafter. The key point, both in [Ran25] and in this work, is that this feature somehow *survives under shortening*.

The exact form of the matrix Φ obviously depends on the choice of basis. We specify how to construct Φ in the so-called "canonical basis" (we use here the terminology of [CMT23b]) of the alternant code $\mathscr{C} = (\mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y}))_{\mathbb{F}_{q^m}}$ we are interested in. This basis is given by

$$\mathcal{B} \stackrel{\text{def}}{=} \{\boldsymbol{y}^{q^u}\boldsymbol{x}^{aq^u} \,:\, u \in [\![0, m-1]\!],\ a \in [\![0, r-1]\!]\}, \tag{15}$$

i.e. it is nothing but the concatenation of the standard basis of the GRS codes involved in (8). We temporarily re-index the variables $X_1, \ldots, X_{rm}$ as $(X_a^{(u)})_{\substack{0 \le a < r \\ 0 \le u < m}}$, and the evaluation map is given by mapping $X_a^{(u)}$ onto $\boldsymbol{y}^{q^u}\boldsymbol{x}^{aq^u}$.

Alternant case. Clearly we have a bunch of elements of $\mathscr{C}_{\mathrm{rel}}(\mathcal{B})$ given by quadratic relations of the kind for $a, b, c, d \in [\![0, r-1]\!]$ and $u, v \in [\![0, m-1]\!]$,

$$\boldsymbol{y}^{q^u}\boldsymbol{x}^{aq^u}\boldsymbol{y}^{q^v}\boldsymbol{x}^{bq^v} - \boldsymbol{y}^{q^u}\boldsymbol{x}^{cq^u}\boldsymbol{y}^{q^v}\boldsymbol{x}^{dq^v} = \mathbf{0},$$

when $aq^u + bq^v = cq^u + dq^v$. In the square distinguishable alternant case, it turns out that these relations apparently generate $\mathscr{C}_{\mathrm{rel}}(\mathcal{B})$, but they are clearly no independent. The approach of [Ran25] is to find a basis of these relations such that the relations can be interpreted as the minors of a certain matrix $\mathbf{\Phi}$. The basis of the quadratic relations is chosen to be the relations of the form

$$\boldsymbol{y}^{q^{e-u}}\boldsymbol{x}^{aq^{e-u}}\boldsymbol{y}^{q^{e-v}}\boldsymbol{x}^{(b+q^v)q^{e-v}} - \boldsymbol{y}^{q^{e-u}}\boldsymbol{x}^{(a+q^u)q^{e-u}}\boldsymbol{y}^{q^{e-v}}\boldsymbol{x}^{bq^v} = \mathbf{0}, \tag{16}$$

where $e = e_{\mathscr{A}}$, where $e_{\mathscr{A}}$ is defined in (10), where u, v range over $[\![0, e]\!]$, a ranges over $[\![0, r-1-q^u]\!]$, b ranges over $[\![0, r-1-q^v]\!]$. These considerations lead to define

$$\boldsymbol{B}_u \stackrel{\text{def}}{=} \begin{pmatrix} X_0^{(0)} & X_1^{(0)} & \cdots & X_{r-1-q^u}^{(0)} \\ X_{q^u}^{(0)} & X_{q^u+1}^{(0)} & \cdots & X_{r-1}^{(0)} \end{pmatrix} \tag{17}$$

$$\boldsymbol{\Phi}_{\mathscr{A}} \stackrel{\text{def}}{=} \left(\boldsymbol{B}_0^{(e)} \middle| \boldsymbol{B}_1^{(e-1)} \middle| \cdots \middle| \boldsymbol{B}_e^{(0)} \right) \tag{18}$$

It is readily seen that all minors $\begin{vmatrix} X_a^{(e-u)} & X_b^{(e-v)} \\ X_{a+q^u}^{(e-u)} & X_{b+q^v}^{(e-v)} \end{vmatrix}$ of $\boldsymbol{\Phi}_{\mathscr{A}}$ correspond to the quadratic relations (16). This is [Ran25, Lem. 7]. The number $f_{\mathscr{A}}$ of columns of $\boldsymbol{\Phi}_{\mathscr{A}}$ is readily seen to be given by

$$f_{\mathscr{A}} = (e_{\mathscr{A}} + 1)r - \frac{q^{e_{\mathscr{A}}+1} - 1}{q - 1}. \tag{19}$$

For the rest of the paper, we assume now

Assumption 1. *We work from now on in characteristic 2. The codes we are interested in are binary. When we take their extension over an extension field, it will be over $\mathbb{F}_{2^m}$.*

The case of binary (squarefree) Goppa codes. When the Goppa polynomial Γ is squarefree, the binary Goppa code $\mathscr{G}(\boldsymbol{x}, \Gamma)$ can be viewed not only as an alternant code $\mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y})$ where $r = \deg \Gamma$ and $\boldsymbol{y} = \frac{1}{\Gamma(\boldsymbol{x})}$ as given by its definition, but also as an alternant code of twice its degree, due to the fact that

$$\mathscr{G}(\boldsymbol{x}, \Gamma) = \mathscr{G}(\boldsymbol{x}, \Gamma^2) = \mathscr{A}_{2r}\left(\boldsymbol{x}, \boldsymbol{y}^2\right).$$

For the first equality see [MS77, p. 342] or [Ran25, Lem. 12]. This allows to define a $2 \times f_{\mathscr{G}}$ matrix $\boldsymbol{\Phi}_{\mathscr{G}}$ which is about twice longer and whose 2×2 minors all belong to $I_2(\mathscr{C})$. More precisely, we use the following equality.

Lemma 2.5 ([Ran25], Proposition 7). *Let $\mathscr{C} \stackrel{\text{def}}{=} \mathscr{G}(\boldsymbol{x}, \Gamma)^\perp$ be the dual of a proper binary Goppa code with square-free Goppa polynomial Γ, and let $\boldsymbol{z} \stackrel{\text{def}}{=} \Gamma(\boldsymbol{x})^{-2}$. Then*

$$\mathscr{C}_{\mathbb{F}_{q^m}} = \bigoplus_{j=0}^{m/2-1} \mathbf{GRS}_{2r}\left(\boldsymbol{x}^{4^j}, \boldsymbol{z}^{4^j}\right),$$

or

$$\mathscr{C}_{\mathbb{F}_{q^m}} = \bigoplus_{j=0}^{(m-1)/2-1} \mathbf{GRS}_{2r}\left(\boldsymbol{x}^{4^j}, \boldsymbol{z}^{4^j}\right) \oplus \mathbf{GRS}_r\left(\boldsymbol{x}^{2^{m-1}}, \boldsymbol{y}^{2^{m-1}}\right),$$

depending on the parity of m.

We may now write variables $(X_i^{(j)})_{i,j}$ accordingly. We assume m even without loss of generality. Then, much like we did in the alternant case, the basis element

$(\boldsymbol{z}\boldsymbol{x}^i)^{4^j}$ is associated with a variable $X_i^{(j)}$, for all integers $0 \leq i < 2r$ and $0 \leq j \leq m/2 - 1$. Finally, if we write for all $0 \leq u \leq \hat{e} \stackrel{\text{def}}{=} \lfloor \log_4(2r-1) \rfloor$,

$$\boldsymbol{B}_u \stackrel{\text{def}}{=} \begin{pmatrix} X_0^{(0)} & X_1^{(0)} & \dots & X_{2r-1-4^u}^{(0)} \\ X_{4^u}^{(0)} & X_{4^u+1}^{(0)} & \dots & X_{2r-1}^{(0)} \end{pmatrix}, \tag{20}$$

and

$$\boldsymbol{\Phi}_{\mathscr{G}} \stackrel{\text{def}}{=} \left(\boldsymbol{B}_0^{(\hat{e})} \middle| \boldsymbol{B}_1^{(\hat{e}-1)} \middle| \dots \middle| \boldsymbol{B}_{\hat{e}}^{(0)} \right), \tag{21}$$

then the 2×2 minors of $\boldsymbol{\Phi}_{\mathscr{G}}$ all belong to $I_2(\mathscr{C})$ [Ran25, Cor. 5]. We easily see that the matrix $\boldsymbol{\Phi}_{\mathscr{G}}$ has length

$$f_{\mathscr{G}} \stackrel{\text{def}}{=} (2\hat{e}+2)r - \frac{4^{\hat{e}+1}-1}{3}. \tag{22}$$

2.5 The Pfaffian modeling

As explained in the introduction the matrix code of quadratic relations of the dual of an alternant or Goppa code contains many low rank quadratic forms. In particular, many of them happen to have rank 2 when considering codes over fields of characteristic 2. This led the authors of [CMT23b] to model the corresponding MinRank problem – consisting in finding matrices of rank 2 in $\mathscr{C}_{\text{mat}}$ – using the Pfaffian modeling described thereafter.

The modeling we are about to write here will apply not only to the original Goppa code but also to its shortening. For the sake of generality, we consider a general code $\mathscr{C}$ of dimension D and fix a basis $\mathcal{B}$ of the code $\mathscr{C}$. We then compute a basis $(\boldsymbol{M}_1, \cdots, \boldsymbol{M}_K)$ of $\mathscr{C}_{\text{mat}}(\mathcal{B})$. Here the $\boldsymbol{M}_i$'s are $D \times D$ antisymmetric matrices. Let $\boldsymbol{M}$ be defined by

$$\boldsymbol{M} = \sum_{i=1}^{K} z_i \boldsymbol{M}_i, \tag{23}$$

where the z_i's are formal variables. We express that the rank of $\boldsymbol{M}$ is less than or equal to 2 by asking that all 4×4 principal minors of $\boldsymbol{M}$ are 0. It is readily seen that if we denote by $m_{i,j}$ the entry of $\boldsymbol{M}$, then the 4×4 minor associated to the rows and columns i, j, k, ℓ is given by

$$\begin{vmatrix} 0 & m_{i,j} & m_{i,k} & m_{i,\ell} \\ m_{i,j} & 0 & m_{j,k} & m_{j,\ell} \\ m_{i,k} & m_{j,k} & 0 & m_{k,\ell} \\ m_{i,\ell} & m_{j,\ell} & m_{k,\ell} & 0 \end{vmatrix} = (m_{i,j}m_{k,l} + m_{i,k}m_{j,l} + m_{i,l}m_{j,k})^2 \tag{24}$$

The term $m_{i,j}m_{k,l} + m_{i,k}m_{j,l} + m_{i,l}m_{j,k}$ is called the Pfaffian corresponding to the principal minor associated to rows and columns i, j, k, ℓ. It will be denoted by $\mathbf{Pf}(\boldsymbol{M}[i,j,k,l])$. Expressing that $\boldsymbol{M}$ is of rank ≤ 2 is equivalent to require that all these Pfaffians are equal to 0, since all odd minors of an antisymmetric matrix

are equal to 0 and all 4×4 minors can be expressed as bilinear combinations of the 4×4 Pfaffians (this follows from the Pfaffian–Binet–Cauchy formula). This leads to the following set of $\binom{D}{4}$ quadratic equations in the z_i's

Modeling 1 (Pfaffian modeling). *The Pfaffian modeling expressing that an antisymmetric matrix*

$$\boldsymbol{M} = (m_{i,j})_{\substack{i \in [\![1,D]\!] \\ j \in [\![1,D]\!]}} = \sum_{i=1}^{K} z_i \boldsymbol{M}_i$$

of size $D \times D$ belonging to the space generated by K antisymmetric matrices $\boldsymbol{M}_1, \cdots, \boldsymbol{M}_K$ is of rank ≤ 2 is given by

Unknowns: $z_1, \cdots, z_K$

Equations: $\binom{D}{4}$ *quadratic equations of the form*

$$m_{i,j}m_{k,l} + m_{i,k}m_{j,l} + m_{i,l}m_{j,k} = 0$$

where the $m_{i,j}$ are linear forms in the z_i's that are specified by (23).

3 Shortening before Looking for Rank 2 Matrices in $\mathscr{C}_{\mathbf{mat}}$

The purpose of this section is to show that we can shorten by a large amount the extended dual code of the alternant/Goppa code in which we want to find rank 2 matrices in the associated matrix code of relations. This decreases significantly the number of unknowns in the algebraic attack. The crucial argument is the determinantal modeling put forward in [Ran25] and which is recalled in §2.4. The determinantal modeling is used in [Ran25] to lower bound the Betti numbers involved in the syzygy distinguisher. It is also used in a crucial way to lower bound those Betti numbers when the dual of the alternant or Goppa code gets shortened before applying the syzygy distinguisher.

3.1 The alternant case

Let $\mathscr{C} = \left(\mathscr{A}_r(\boldsymbol{x}, \boldsymbol{y})^{\perp}\right)_{\mathbb{F}_{q^m}}$ be the extended dual of the alternant code we want to distinguish. Let us assume that we shorten $\mathscr{C}$ in a set J of s positions and let $\mathscr{C}_s$ be the resulting code. Without loss of generality, we may assume after permuting the positions that these are the last s positions of the code. We also order the set J and denote for $j \in [\![1, s]\!]$ by $\mathscr{C}_j$ the code $\mathscr{C}$ shortened in the last j positions in J. Let $\boldsymbol{\Phi}$ be the $2 \times f_{\mathscr{A}}$ matrix defined in §2.4 whose 2×2 minors all belong to $I_2(\mathscr{C})$. We denote by k the dimension of $\mathscr{C}$ and by $\boldsymbol{G}$ be the generator matrix of $\mathscr{C}$. We assume that this generator matrix is chosen in such a way that the first $k - j$ rows of $\boldsymbol{G}$ form a basis of $\mathscr{C}_j$ for every $j \in J$ and that it is in systematic form in its k last positions. We denote by $\boldsymbol{G}_j$ the generator matrix

of $\mathscr{C}_j$ formed by the $k-j$ first rows of $\boldsymbol{G}$ and throwing out the last j columns. For the assumption on the systematic part, this assumption can be done at the cost of permuting the positions, for the rest we implicitly made the following assumption

Assumption 2.

$$\dim \mathscr{C}_s = \dim \mathscr{C} - s.$$

It may happen that the dimension is bigger than this when we have for some $j \in [\![0, s-1]\!]$ that $\mathscr{C}_j = \mathscr{C}_{j+1}$ (where $\mathscr{C}_0 \stackrel{\text{def}}{=} \mathscr{C}$). We may in this case just reason on the subset J' of positions of J which are such that when we order J' again and define $\mathscr{C}'_j$ as $\mathscr{C}$ shortened in the last j positions of J' we have for all $j \in [\![0, |J'|-1]\!]$, $\dim \mathscr{C}'_{j+1} = \dim \mathscr{C}'_j - 1$ and $\dim \mathscr{C}'_{s'} = \dim \mathscr{C} - s'$ where $s' \stackrel{\text{def}}{=} |J'|$.

We denote the rows of $\boldsymbol{G}$ by $\boldsymbol{c}_1, \cdots, \boldsymbol{c}_k$ and consider as in §2.3 the evaluation map

$$\begin{aligned} \boldsymbol{e}_{\boldsymbol{G}} : \mathcal{R} &\to \mathbb{F}_{q^m}^n \\ f &\mapsto f(\boldsymbol{c}_1, \cdots, \boldsymbol{c}_k) \end{aligned}$$

where $\mathcal{R} = \mathbb{F}_{q^m}[X_1, \cdots, X_k]$ is the polynomial ring of the entries of $\boldsymbol{\Phi}$. Recall that $I_2(\mathscr{C})$ is the kernel of $\boldsymbol{e}_{\boldsymbol{G}}$ restricted to the homogeneous polynomials of degree 2 in $\mathcal{R}$. We let $V_{\boldsymbol{\Phi}}$ be the $\mathbb{F}_{q^m}$-vector subspace of $\mathcal{R}^2$ generated by the columns of $\boldsymbol{\Phi}$. It is readily seen that (this is just a minor variation on [Ran25, Prop. 4, 1.] whose first item is this proposition in the case $s = 1$)

Proposition 3.1. *For $j \in [\![1, s]\!]$ we let $\mathcal{R}_j = \mathbb{F}_{q^m}[X_1, \cdots, X_{k-j}]$. Let $V_j \stackrel{\text{def}}{=} V_{\boldsymbol{\Phi}} \cap \mathcal{R}_j^2$. We let $\boldsymbol{\Phi}_j$ be a matrix whose columns generate V_j. All 2×2 minors of $\boldsymbol{\Phi}_j$ belong to $I_2(\mathscr{C}_j)$.*

Proof. Since $V_j \subseteq V_{\boldsymbol{\Phi}}$ and by the multilinearity of the determinant, any 2×2 minor of $\boldsymbol{\Phi}_j$ is a linear combination of minors of $\boldsymbol{\Phi}$. The resulting polynomial P evaluates to $\boldsymbol{0}$ by $\boldsymbol{e}_{\boldsymbol{G}}$. Since it also belongs to $\mathcal{R}_j$ by definition of V_j and is a homogeneous polynomial of degree 2, we have that $P \in I_2(\mathscr{C}_j)$. □

From now on we make an arbitrary choice for $\boldsymbol{\Phi}_1, \cdots, \boldsymbol{\Phi}_s$. Basically it is also proved in [Ran25] that (to keep the paper self-contained we also give a proof in Appendix A)

Lemma 3.2. *We have*

$$\dim V_j \geq \dim V_{j-1} - 1, \ \forall j \in [\![1, s]\!] \tag{25}$$

$$\dim V_j \geq \dim V_{\boldsymbol{\Phi}} - j \tag{26}$$

where $V_0 \stackrel{\text{def}}{=} V_{\boldsymbol{\Phi}}$. If we let f_j be the dimension of V_j for any $j \in [\![1, s]\!]$ then we have that $f_j \geq f_{\mathscr{A}} - j$, where $f_{\mathscr{A}} \stackrel{\text{def}}{=} (e_{\mathscr{A}}+1)r - \frac{q^{e_{\mathscr{A}}+1}-1}{q-1}$ with $e_{\mathscr{A}} \stackrel{\text{def}}{=} \lfloor \log_q(r-1) \rfloor$.

These lemmas imply that the matrices $\mathbf{\Phi}_j$ contain at least $f_{\mathscr{A}} - j$ columns, which gives many quadratic relations which are in $I_2(\mathscr{C}_j)$ by taking all 2×2 minors of this matrix. The polar form associated to such a quadratic form is of rank at most 4. It turns out that we can find matrices of rank 2 associated to the determinant of a well-chosen 2×2 matrix with entries in V_j as long as j is sufficiently small. Basically the approach is to show that we can find a 2×2 minor of $\mathbf{\Phi}_j$ of the form $\begin{vmatrix} a & b \\ b & c \end{vmatrix}$. For this we consider for $j \in [\![1, s]\!]$ the two subspaces V_j^1 and V_j^2 of $\mathcal{R}_j$ as the projection of V_j on the first entry or on the second entry. In other words if we let for $i \in \{1, 2\}$, the projection

$$\begin{aligned} \pi_i : \mathcal{R} \times \mathcal{R} : &\to \mathcal{R} \\ (x_1, x_2) &\mapsto x_i. \end{aligned}$$

We have $V_j^i = \pi_i(V_j)$. The crucial lemma is now

Lemma 3.3. *Let* $\tilde{e}_{\mathscr{A}} \stackrel{\text{def}}{=} \lfloor \log_q \frac{r}{2} \rfloor$ *and* $s_{\mathscr{A}} \stackrel{\text{def}}{=} (\tilde{e}_{\mathscr{A}} + 1)r - 2\frac{q^{\tilde{e}_{\mathscr{A}}+1}-1}{q-1}$. *We have for any* $j \in [\![0, s_{\mathscr{A}}]\!]$,

$$\dim \left(V_j^1 \cap V_j^2 \right) \geq s_{\mathscr{A}} - j.$$

This result which is proved in Appendix A shows that we can shorten $\mathscr{C}$ up to $s = s_{\mathscr{A}} - 1$ positions and derive from the existence of a non-zero element x belonging to both V_s^1 and V_s^2 that one of the two cases happens

- We have a pair of elements in V_s which are of the form $\begin{pmatrix} a \\ x \end{pmatrix}$ and $\begin{pmatrix} x \\ b \end{pmatrix}$
- $\begin{pmatrix} x \\ x \end{pmatrix}$ belongs to V_s.

The first case implies on the spot that we have a matrix of rank 2 in the matrix code of relationships $\mathscr{C}_{\text{mat}}(\mathcal{B}_s)$ in characteristic 2 where $\mathcal{B}_s$ is a basis of $\mathscr{C}_s$. It is the polar form of the quadratic relationship associated to the minor $g \stackrel{\text{def}}{=} \begin{vmatrix} a & x \\ x & b \end{vmatrix}$ belonging to $I_2(\mathscr{C}_s)$. Let us rule out now the second case.

Lemma 3.4. *We have for any* $s \in [\![0, s_{\mathscr{A}}]\!]$ *that*

$$\left\{ x \in \mathcal{R}_s : \begin{pmatrix} x \\ x \end{pmatrix} \in V_s \right\} = \{0\}.$$

This result is proved in Appendix A. With these lemmas at hand, we can now prove our main result of this subsection which is

Theorem 3.5. *Let* $\tilde{e}_{\mathscr{A}} \stackrel{\text{def}}{=} \lfloor \log_q \frac{r}{2} \rfloor$ *and* $s_{\mathscr{A}} \stackrel{\text{def}}{=} (\tilde{e}_{\mathscr{A}} + 1)r - 2\frac{q^{\tilde{e}_{\mathscr{A}}+1}-1}{q-1}$. *Let* $\mathscr{C} = \left(\mathscr{A}_r (\boldsymbol{x}, \boldsymbol{y})^{\perp} \right)_{\mathbb{F}_{q^m}}$ *be a proper alternant code where* $\mathbb{F}_{q^m}$ *is a finite field of characteristic* 2. *For any* $s \in [\![0, s_{\mathscr{A}} - 1]\!]$, *the matrix code of quadratic relations of* $\mathscr{C}_s$ *contains a rank* 2 *matrix, where* $\mathscr{C}_s$ *is* $\mathscr{C}$ *shortened in* s *positions.*

Proof. We can use Lemma 3.3 to show that for such a value s, $V_s^1 \cap V_s^2 \neq \{0\}$. Let $x \neq 0$ being in this intersection. From Lemma 3.4 we know that there exist two elements in V_j of the form $\begin{pmatrix} a \\ x \end{pmatrix}$ and $\begin{pmatrix} x \\ b \end{pmatrix}$. The minor $g \stackrel{\text{def}}{=} \begin{vmatrix} a & x \\ x & b \end{vmatrix}$ belongs to $I_2(\mathscr{C}_s)$. Since $\boldsymbol{e}_{\boldsymbol{G}_s}(g) = \boldsymbol{a}\star\boldsymbol{b} - \boldsymbol{x}\star\boldsymbol{x}$ with $\boldsymbol{a} \stackrel{\text{def}}{=} \boldsymbol{e}_{\boldsymbol{G}_s}(a)$, $\boldsymbol{b} \stackrel{\text{def}}{=} \boldsymbol{e}_{\boldsymbol{G}_s}(b)$, $\boldsymbol{x} \stackrel{\text{def}}{=} \boldsymbol{e}_{\boldsymbol{G}_s}(x)$, we deduce that $\mathscr{C}_{\text{rel}}(\mathcal{B}_s)$ contains the quadratic relation $\boldsymbol{a}\star\boldsymbol{b} - \boldsymbol{x}\star\boldsymbol{x}$ where $\mathcal{B}_s$ is the basis of $\mathscr{C}_s$ formed by the $k-s$ rows of $\boldsymbol{G}_s$. This gives a matrix of rank 2 in the matrix code of relationships $\mathscr{C}_{\text{mat}}(\mathcal{B}_s)$ in characteristic 2. □

This theorem shows that we can shorten $\mathscr{C}$ in $s_{\mathscr{A}} - 1$ positions and still have matrices of rank 2 in the matrix code of quadratic relationships of $\mathscr{C}_s$. Experimentally, it turns out that this lower bound is tight. We report on Table 3 the values of $s_{\mathscr{A}}$ for different set of parameters, together with

- the dimension of the ideal $\mathcal{P}_{\mathscr{A}}$ of the Pfaffian modeling for an alternant code before shortening, *i.e.* taking into acount all quadratic relations in $I_2(\mathscr{C})$;
- the dimension of the ideal $\mathcal{P}_{\text{det}}$ of the Pfaffian modeling for rank-2 matrices in the matrix code generated by the matrices of the polar forms of the 2×2 minors of $\boldsymbol{\Phi}$. As these minors are a subset of $I_2(\mathscr{C})$, we expect the dimension of this ideal to be at most equal to that of $\mathcal{P}_{\mathscr{A}}$;
- the number of positions s_0 at which we actually shortened the code in practice before writing the Pfaffian modeling and solving it. We expect this value to be at least $s_{\mathscr{A}} - 1$. The following table shows that it was exactly $s_{\mathscr{A}} - 1$.

r	m	$s_{\mathscr{A}}$	$\dim \mathcal{P}_{\text{det}}$	$\dim \mathcal{P}_{\mathscr{A}}$	s_0
3	10	1	1	1	0
4	10	2	2	2	1
5	10	4	4	4	3
6	11	6	6	6	5
8	12	10	10	10	9
9	12	13	13	13	N/A

Table 3: Dimension of the Pfaffian ideal for alternant codes.

Table 3 shows that the dimension of the Pfaffian ideal for alternant codes always coincides with $s_{\mathscr{A}}$.

3.2 The binary Goppa case

Much of what is written in the last section also works when $\mathscr{C} = \left(\mathscr{G}(\boldsymbol{x}, \Gamma)^{\perp}\right)_{\mathbb{F}_{2^m}}$. We define the set V_j's in a similar way. Lemma 3.3 holds in the same way: it just relies on the existence of a matrix $\boldsymbol{\Phi}$ for which Proposition 3.1 applies and this is the case for the $\boldsymbol{\Phi}$ given in (21) in §2.4. The only difference with the previous case is

Lemma 3.6.

$$\begin{aligned}\dim(V_0^1 \cap V_0^2) &= s_{\mathscr{G}} \\ \textit{where } s_{\mathscr{G}} &\stackrel{\text{def}}{=} 2\left((\tilde{e}_{\mathscr{G}}+1)r - \frac{4^{\tilde{e}_{\mathscr{G}}+1}-1}{3}\right) \\ \tilde{e}_{\mathscr{G}} &\stackrel{\text{def}}{=} \lfloor \log_4 r \rfloor.\end{aligned}$$

Proof. The dimensions of the subspaces we are interested in do not depend on the chosen generator matrix. It is convenient here to consider the canonical basis (15). In this basis the matrix $\mathbf{\Phi}$ is given by

$$\begin{aligned}\boldsymbol{B}_u &\stackrel{\text{def}}{=} \begin{pmatrix} X_0^{(0)} & X_1^{(0)} & \dots & X_{2r-1-4^u}^{(0)} \\ X_{4^u}^{(0)} & X_{4^u+1}^{(0)} & \dots & X_{2r-1}^{(0)} \end{pmatrix} \\ \mathbf{\Phi} &\stackrel{\text{def}}{=} \left(\boldsymbol{B}_0^{(\hat{e})} \middle| \boldsymbol{B}_1^{(\hat{e}-1)} \middle| \dots \middle| \boldsymbol{B}_e^{(0)}\right) \\ \text{where } \hat{e} &\stackrel{\text{def}}{=} \lfloor \log_4(2r-1) \rfloor.\end{aligned}$$

From this form it is readily seen that $\dim(V_0^1 \cap V_0^2)$ is given by the number N of variables that appear both in the first row and the second row of $\mathbf{\Phi}$. We clearly have

$$\begin{aligned}\dim(V_0^1 \cap V_0^2) = N &= \sum_{u=0}^{\hat{e}} \max\left(2r - 2\cdot 4^u, 0\right) \\ &= 2\sum_{u=0}^{\tilde{e}_{\mathscr{G}}} (r - 4^u) \\ &= 2\left((\tilde{e}_{\mathscr{G}}+1)r - \frac{4^{\tilde{e}_{\mathscr{G}}+1}-1}{3}\right). \end{aligned} \tag{27}$$

□

The same arguments used for alternant codes apply now, and we derive from it that

Theorem 3.7. *Let* $\tilde{e}_{\mathscr{G}} \stackrel{\text{def}}{=} \lfloor \log_4 r \rfloor$ *and* $s_{\mathscr{G}} \stackrel{\text{def}}{=} 2\left((\tilde{e}_{\mathscr{G}}+1)r - \frac{4^{\tilde{e}_{\mathscr{G}}+1}-1}{3}\right)$. *Let* $\mathscr{C} = \left(\mathscr{G}(\boldsymbol{x}, \Gamma)^{\perp}\right)_{\mathbb{F}_2^m}$ *where* Γ *is a square free polynomial in* $\mathbb{F}_{2^m}[X]$ *of degree* r. *For any* $s \in [\![0, s_{\mathscr{G}} - 1]\!]$, *the matrix code of quadratic relations of* $\mathscr{C}_s$ *contains a rank* 2 *matrix, where* $\mathscr{C}_s$ *is* $\mathscr{C}$ *shortened in* s *positions.*

We again report in Table 4 the dimension of the Pfaffian ideal $\mathcal{P}_{\text{det}}$ associated to the 2×2 minors of $\mathbf{\Phi}_{\mathscr{G}}$, and the dimension of the Pfaffian ideal $\mathcal{P}_{\mathscr{G}}$ when taking into account all relations in $I_2(\mathscr{C})$. Some dimensions were however not computable for all sets of parameters.

r	m	$s_{\mathscr{G}}$	$\dim \mathcal{P}_{\det}$	$\dim \mathcal{P}_{\mathscr{G}}$	s_0
2	8	2	2	3	1
3	8	4	4	6	**4**
4	8	6	6	N/A	**7**
5	8	10	10	N/A	**11**
6	9	14	14	N/A	**15**
7	9	18	N/A	N/A	**19**
8	11	22	N/A	N/A	**22**

Table 4: Dimension of the Pfaffian ideal for Goppa codes.

On Table 4, bold numbers indicate that we were able to shorten more than the theoretical bound $s_{\mathscr{G}} - 1$ suggests. Contrary to the alternant case, we observe $\dim \mathcal{P}_{\det} < \dim \mathcal{P}_{\mathscr{G}}$. Note that $\dim \mathcal{P}_{\mathscr{G}} = 3$ when $r = 2$ was already explained in details in [Mor25]. The inequality being strict in general comes from the fact that some quadratic forms in $I_2(\mathscr{C})$ simply cannot be expressed as linear combinations of 2×2 minors of $\mathbf{\Phi}_{\mathscr{G}}$ and its conjugates.

4 Squares in the Pfaffian Modeling

If we perform a Gröbner basis computation for the Pfaffian modeling we noticed that there are quadratic polynomials which are squares f^2 in the corresponding ideal. This happens regardless of whether we shorten or not. Obtaining such equations is really helpful. Indeed if we replace f^2 by the linear form f, it does not change the associated variety, but this has the advantage to speed up significantly the computation since we can eliminate one variable with each (independent) linear form and even more importantly this ensures that the computation of the Gröbner basis stays at degree 2. Such squares can be computed directly without performing a Gröbner basis computation by solving a linear system. Indeed, the generators of our Pfaffian ideal are all polynomials with binary coefficients, so detecting degree 2 polynomials which are squares amounts to find quadratic polynomials with only quadratic monomials which are squares, *i.e.* of the form x_i^2.

Let us explain now why there are squares in the Pfaffian modeling before shortening. We will explain what is going on for a specific example. Let us take $\mathscr{C} = \left(\mathscr{A}_{\boldsymbol{x}}(\boldsymbol{y}, r)^{\perp}\right)_{F_{q^m}}$ with $r = m = 4$ and $q = 2$ and consider the associated Pfaffian ideal. This code is square-distinguishable and the dimension K of $I_2(\mathscr{C})$ is 36 in this case. The fact that there are squares in the Pfaffian ideal does not depend on the basis chosen for $\mathscr{C}$. It will be helpful to consider the canonical basis for $\mathscr{C}$ which is $\mathcal{B} = \left\{\boldsymbol{y}^{q^i}\boldsymbol{x}^{aq^i},\ i \in [\![0, m-1]\!],\ a \in [\![0, r-1]\!]\right\}$. The code of quadratic relationships $\mathscr{C}_{\text{rel}}\mathcal{B}$ is generated by all relations of the form

$$(\boldsymbol{y}\boldsymbol{x}^i)^{q^u} \star (\boldsymbol{y}\boldsymbol{x}^j)^{q^v} = (\boldsymbol{y}\boldsymbol{x}^k)^{q^u} \star (\boldsymbol{y}\boldsymbol{x}^l)^{q^v}, \tag{28}$$

where $0 \leq i, j, k, l < r$ and $0 \leq u, v < m$ are integers such that $iq^u + jq^v = kq^u + lq^v$. There are 36 independent relationships of this kind. To construct a basis of $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ it will be helpful to choose a basis formed only by rank 2 and 4 matrices that correspond to a subset of quadratic relationships of the form (28). Quadratic relationships of this kind where $k = l$ and $u = v$ have a polar form of rank 2 where the nonzero part of the matrix correspond to rows and columns corresponding to $\boldsymbol{y}^{q^u}\boldsymbol{x}^{iq^u}$ and $\boldsymbol{y}^{q^u}\boldsymbol{x}^{jq^u}$ which is of the form

$$\begin{array}{c} \\ \boldsymbol{y}\boldsymbol{x}^{iq^u} - \\ \boldsymbol{y}\boldsymbol{x}^{kq^u} - \\ \boldsymbol{y}\boldsymbol{x}^{jq^u} - \end{array} \begin{array}{c} \begin{array}{ccc} \boldsymbol{y}\boldsymbol{x}^{iq^u} & \boldsymbol{y}\boldsymbol{x}^{kq^u} & \boldsymbol{y}\boldsymbol{x}^{jq^u} \\ | & | & | \end{array} \\ \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \end{array}$$

The other quadratic relations have a polar form whose nonzero part is a 4×4 matrix of rank 2 given by

$$\begin{array}{c} \\ \boldsymbol{y}\boldsymbol{x}^{iq^u} - \\ \boldsymbol{y}\boldsymbol{x}^{kq^u} - \\ \boldsymbol{y}\boldsymbol{x}^{lq^v} - \\ \boldsymbol{y}\boldsymbol{x}^{jq^v} - \end{array} \begin{array}{c} \begin{array}{cccc} \boldsymbol{y}\boldsymbol{x}^{iq^u} & \boldsymbol{y}\boldsymbol{x}^{kq^u} & \boldsymbol{y}\boldsymbol{x}^{lq^v} & \boldsymbol{y}\boldsymbol{x}^{jq^v} \\ | & | & | & | \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

We can choose a basis $\{\boldsymbol{M}_1, \cdots, \boldsymbol{M}_{36}\}$ of $\mathscr{C}_{\mathrm{mat}}\mathcal{B}$ with a maximum number of rank 2 matrices. Let $\boldsymbol{M} \stackrel{\text{def}}{=} \sum_{i=1}^{36} z_i \boldsymbol{M}_i$. It is given in Figure 1. We have put in red all variables z_i which after writing the corresponding Pfaffian modeling are such that their square z_i^2 is in the liner span of these quadratic polynomials. It is readily verified on an example why this is the case. Consider the variable z_3 for instance. Consider the Pfaffian corresponding to the 4 rows and columns containing a z_3 term. It corresponds to the 4×4 matrix $\boldsymbol{P}$ which contains the support of the $\boldsymbol{M}_3$ matrix. This Pfaffian is the square root of the following minor

$$\begin{vmatrix} 0 & z_1 & 0 & z_3 \\ z_1 & 0 & z_3 & z_4 \\ 0 & z_3 & 0 & 0 \\ z_3 & z_4 & 0 & 0 \end{vmatrix}$$

The corresponding Pfaffian is equal to $z_1 \times 0 + 0 \times z_4 + z_3^2 = z_3^2$. We have indicated in red the zeros that are responsible for the fact that this Pfaffian is actually a square. $\boldsymbol{M}_3$ corresponds to the following quadratic relationship:

$$\boldsymbol{y} \star \boldsymbol{y}^2\boldsymbol{x}^2 = \boldsymbol{y}\boldsymbol{x}^2 \star \boldsymbol{y}^2.$$

Fig. 1: Matrix $\boldsymbol{M}$ corresponding to $r = m = 4$ where the variables in red appear as squares in the Pfaffian modeling.

$$\boldsymbol{M} = \left(\begin{array}{cccc|cccc|cccc|cccc}
0 & 0 & z_1 & z_2 & 0 & z_3 & z_4 & z_5 & 0 & 0 & 0 & 0 & 0 & 0 & z_6 & z_7 \\
0 & 0 & z_2 & z_8 & 0 & z_9 & z_{10} & z_{11} & 0 & 0 & 0 & 0 & z_6 & z_7 & z_{12} & z_{13} \\
z_1 & z_2 & 0 & 0 & z_3 & z_4 & z_5 & 0 & 0 & 0 & 0 & 0 & z_{12} & z_{13} & z_{14} & z_{15} \\
z_2 & z_8 & 0 & 0 & z_9 & z_{10} & z_{11} & 0 & 0 & 0 & 0 & 0 & z_{14} & z_{15} & 0 & 0 \\
\hline
0 & 0 & z_3 & z_9 & 0 & 0 & z_{16} & z_{17} & 0 & z_{18} & z_{19} & z_{20} & 0 & 0 & 0 & 0 \\
z_3 & z_9 & z_4 & z_{10} & 0 & 0 & z_{17} & z_{21} & 0 & z_{22} & z_{23} & z_{24} & 0 & 0 & 0 & 0 \\
z_4 & z_{10} & z_5 & z_{11} & z_{16} & z_{17} & 0 & 0 & z_{18} & z_{19} & z_{20} & 0 & 0 & 0 & 0 & 0 \\
z_5 & z_{11} & 0 & 0 & z_{17} & z_{21} & 0 & 0 & z_{22} & z_{23} & z_{24} & 0 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 0 & z_{18} & z_{22} & 0 & 0 & z_{25} & z_{26} & 0 & z_{27} & z_{28} & z_{29} \\
0 & 0 & 0 & 0 & z_{18} & z_{22} & z_{19} & z_{23} & 0 & 0 & z_{26} & z_{30} & 0 & z_{31} & z_{32} & z_{33} \\
0 & 0 & 0 & 0 & z_{19} & z_{23} & z_{20} & z_{24} & z_{25} & z_{26} & 0 & 0 & z_{27} & z_{28} & z_{29} & 0 \\
0 & 0 & 0 & 0 & z_{20} & z_{24} & 0 & 0 & z_{26} & z_{30} & 0 & 0 & z_{31} & z_{32} & z_{33} & 0 \\
\hline
0 & z_6 & z_{12} & z_{14} & 0 & 0 & 0 & 0 & 0 & 0 & z_{27} & z_{31} & 0 & 0 & z_{34} & z_{35} \\
0 & z_7 & z_{13} & z_{15} & 0 & 0 & 0 & 0 & z_{27} & z_{31} & z_{28} & z_{32} & 0 & 0 & z_{35} & z_{36} \\
z_6 & z_{12} & z_{14} & 0 & 0 & 0 & 0 & 0 & z_{28} & z_{32} & z_{29} & z_{33} & z_{34} & z_{35} & 0 & 0 \\
z_7 & z_{13} & z_{15} & 0 & 0 & 0 & 0 & 0 & z_{29} & z_{33} & 0 & 0 & z_{35} & z_{36} & 0 & 0
\end{array}\right)$$

It is insightful to give for the corresponding 4×4 matrices the index of the basis elements which index the rows and columns of $\boldsymbol{P}$:

$$\begin{array}{r} \\ \boldsymbol{y}- \\ \boldsymbol{y}\boldsymbol{x}^2- \\ \boldsymbol{y}^2- \\ \boldsymbol{y}^2\boldsymbol{x}^2- \end{array} \begin{array}{c} \begin{array}{cccc} \boldsymbol{y} & \boldsymbol{y}\boldsymbol{x}^2 & \boldsymbol{y}^2 & \boldsymbol{y}^2\boldsymbol{x}^2 \end{array} \\ \left(\begin{array}{cccc} 0 & z_1 & 0 & z_3 \\ z_1 & 0 & z_3 & z_4 \\ 0 & z_3 & 0 & 0 \\ z_3 & z_4 & 0 & 0 \end{array}\right) \end{array}$$

The reason why there is a 0 at index $(\boldsymbol{y}^2, \boldsymbol{y}^2\boldsymbol{x}^2)$ is that there is no quadratic relationship of the form (28) involving the term $\boldsymbol{y}^2 \star \boldsymbol{y}^2\boldsymbol{x}^2$. The same explanation holds for the 0 at index $(\boldsymbol{y}^2, \boldsymbol{y}^2\boldsymbol{x}^2)$: here there is no quadratic relationship of the form (28) involving the term $\boldsymbol{y} \star \boldsymbol{y}^2$. Similar explanations can be given for all the variables that appear in red in Figure 1. For larger values of r, the fact that certain terms do not appear in (28) only explain the vast majority of the squares that appear in the linear span of the Pfaffian modeling but not all of them. In any case counting such Pfaffians gives an easy lower bound on the number of independent squares appearing in the linear span of the Pfaffian modeling.

All this explains the existence of perfect squares in the Pfaffian modeling, at least in the square-distinguishable regime before shortening. It turns out that many perfect squares remain visible after shortening, even in a regime where the shortening is not square distinguishable. This is what we observed in practice, notably when solving TII challenges.

5 The Algorithm of the Attack

Let $\mathscr{G} = \mathscr{G}(\boldsymbol{x}, \Gamma)$ be a binary Goppa code of extension degree m with square-free Goppa polynomial Γ. We let $\mathscr{C} = (\mathscr{G}^{\perp})_{\mathbb{F}_{2^m}}$ be the extended dual code over $\mathbb{F}_{2^m}$. Throughout this section, we write $\boldsymbol{y} \stackrel{\text{def}}{=} \Gamma(\boldsymbol{x})^{-1}$. We describe here how to recover a pair $(\boldsymbol{x}', \Gamma')$ such that $\mathscr{G} = \mathscr{G}(\boldsymbol{x}', \Gamma')$ using Algorithm 1. Roughly speaking this algorithm computes $t = \lceil e_{\mathscr{G}}/3 \rceil$ different shortened codes $\mathscr{C}_s^1, \cdots, \mathscr{C}_s^t$ with respect to s positions each time (where s is well chosen) and for each of those shortened codes $\mathscr{C}_s^i$ computes a rank 2 matrix belonging to $\mathscr{C}_{\text{mat}}(\mathcal{B}_s^i)$ (where $\mathcal{B}_s^i$ is an arbitrary basis of $\mathscr{C}_s^i$). The resulting t rank 2 matrices yield rank 2 matrices in the matrix code of relations of the original code $\mathscr{C}$, for free. Due to them being computed from different shortenings, we can treat them as random independent rank 2 matrices in the matrix code of $\mathscr{C}$. A generalization of the attack on binary Goppa codes of degree $r = 2$ obtained in [Mor25] then finishes the job.

5.1 Outline of the attack

The outline of the algorithm is the following one:

Algorithm 1 Attack

1: $\mathcal{E} \leftarrow \emptyset$
2: **for** $i = 1$ to t **do**
3: Choose a subset I_i of s positions of $[\![1, n]\!]$ ▷ n is the length of $\mathscr{C}$
4: $\mathscr{C}_s^i \leftarrow \mathbf{Sh}_{I_i}(\mathscr{C})$
5: $\mathcal{M} \leftarrow$ Rank2Matrix$(\mathscr{C}_s^i)$ ▷ Returns a rank 2 matrix in $\mathscr{C}_{\text{mat}}(\mathcal{B}_s^i)$ where $\mathcal{B}_s^i$ is an arbitrary basis of $\mathscr{C}_s^i$
6: $\mathcal{E} \leftarrow \mathcal{E} \cup \mathcal{M}$
7: Use the t rank 2 matrices of $\mathcal{E}$ to reconstruct a basis $\mathcal{B}$ of some $\mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$ for some j ▷ See §5.3
8: Apply [SS92] or [Mor25] to recover a support $\boldsymbol{x}$ and a Goppa polynomial Γ from $\mathcal{B}$
9: **return** $(\boldsymbol{x}, \Gamma)$

5.2 Finding a rank 2 matrix in $\mathscr{C}_s^i$

We explain here how the function Rank2Matrix$(\mathscr{C}_s^i)$ works. Note that what we are going to explain also applies to the case where $\mathscr{G}$ is a binary alternant code. Basically this is done through the following steps:

- Compute a basis $\mathcal{B}_s^i$ of $\mathscr{C}_s^i$ together with a basis $\boldsymbol{M}_1, \ldots, \boldsymbol{M}_K$ of $\mathscr{C}_{\text{mat}}(\mathcal{B}_s^i)$, apply Modeling 1 to $\mathscr{C}_s$ and specialize one variable to 1.
- Solve the relevant algebraic system by
  1. Computing a Gröbner basis at degree 2 of the relevant system.
  2. Finding squares in the linear span of quadratic equations and replace them by their square root.

3. Finding a Gröbner basis at degree 2 of the new system which turns out to be a Gröbner basis of the modified system.
4. Performing a change of order [FGLM93] to get a Gröbner basis for the lexicographical order to output a matrix of rank 2 in $\mathscr{C}_{\mathrm{mat}}(\mathcal{B}_s^i)$.

In the alternant case we choose $s = s_{\mathscr{A}} - 1$ and in the Goppa case s will be just slightly bigger than $s_{\mathscr{G}} - 1$. We adjusted this value by letting s grow, starting at $s_{\mathscr{G}} - 1$ until the Pfaffian modeling for $\mathscr{C}_s$ has dimension 0 after specialization of one variable to 1.

Algorithm 2 Function returning a rank 2 matrix in $\mathscr{C}_{\mathrm{mat}}(\mathscr{B})$ where $\mathscr{B}$ is a basis of $\mathscr{C}$

function RANK2MATRIX($\mathscr{C}$)
  Compute Pfs $= \{\mathbf{Pf}(\boldsymbol{M}[i,j,k,l]) \mid 1 \leq i < j < k < l \leq rm - s\}$; ▷ $\boldsymbol{M}$ is the matrix of Modeling 1 applied to $\mathscr{C}$.
  Compute $\mathscr{G}_1 =$ GRÖBNERBASIS(Pfs, 2); ▷ This just consists in echelonizing the quadratic system, obtaining linear equations and eliminate as many variables as we have obtained linear equations and then echelonize the obtained quadratic system again.
  Compute $\mathscr{S}$ a basis of $\{f \in \langle \mathcal{G}_1 \rangle_{\mathbb{F}_2} \mid \exists g,\ f = g^2\}$;
  $\mathscr{L} \leftarrow \{\sqrt{f} \mid f \in \mathscr{S}\}$;
  Compute $\mathscr{G}_2 =$ GRÖBNERBASIS($\mathscr{L} \cup \mathscr{G}_1, 2$) ; ▷ We eliminate $|\mathscr{L}|$ variables by using the linear equations of $\mathscr{L}$ and echelonize the system.
  $\mathscr{G}_{\mathrm{lex}} \leftarrow$ Gröbner basis in lexicographical order obtained by $\mathscr{G}_2$;
  $\boldsymbol{B} \leftarrow$ SOLVE($\mathscr{G}_{\mathrm{lex}}$);
return $\boldsymbol{B}$ ▷

A few remarks can be made here.

Remark 5.1.
- In all our experiments, Algorithm 2 did not return only one rank 2 matrix, but m. These m matrices were however all the same up to some conjugation operation as we explain in Appendix C. We thus need (and must) keep only one of these m matrices.
- The total number of Pfaffians is equal to $\binom{rm-s}{4}$. We aim at attacking high-rate Goppa codes, especially square-distinguishable Goppa codes. Such codes are such that the dimension K of $I_2(\mathscr{C}_s)$ remains relatively small, leading to

$$\binom{rm-s}{4} \geq \binom{K+1}{2}. \tag{29}$$

  This inequality allows us to avoid computing all Pfaffians.
- Due to the specialization, Step 3 already produces a certain number n_1 of linear equations. The Gröbner basis computation at this step uses these linear forms to eliminate variables. As a result, the number of quadratic equations produced at this step cannot exceed $\binom{K-n_1+1}{2}$.

– Step 4 allows to find n_c additional linear equations. Experimentally, the quantities m, K, n_1, n_c always satisfy the equation

$$n_1 + n_c = K - m. \tag{30}$$

Incorporating the new linear equations, we experimentally obtain a radical ideal generated at degree 2. This is why we stop at Step 6: we expect $\mathscr{G}_2$ to be a Gröbner basis of the radical of the specialized Pfaffian modeling.

Proposition 5.2. *Assume that Algorithm 2 indeed returns m solutions, as noticed in Remark 5.1. Also, assume we are in the regime where Inequality (29) holds. Then the complexity of Algorithm 2 is upper-bounded by*

$$\kappa = \mathcal{O}\left(\binom{rm-s}{4}^{\omega}\right), \tag{31}$$

where ω is the exponent of linear algebra.

Proof. To begin with, note that since the FGLM algorithm here has complexity $\mathcal{O}\left(m^3(K-n_1-n_c)\right)$ (which is $\mathcal{O}\left(m^4\right)$ if we further assume (30)). The bottleneck of Algorithm 2 therefore lies at Step 3, which amounts to echelonizing the Macaulay matrix representing the Pfaffians, deducing linear equations, and use the linear equations to eliminate variables. Again, the bottleneck here is the echelonization. The Macaulay matrix has (at most) $\binom{rm-s}{4}$ rows, which exceeds the number of columns by (29). □

As we explained in Remark 5.1, we do not necessarily need to compute all the Pfaffians when (29) holds. The rank of the Macaulay matrix cannot exceed the number of columns, which is $\binom{K+1}{2}$. Therefore, we should in theory be able to pick $\binom{K+1}{2}$ subsets $\{i,j,k,l\} \subset [\![1, rm-s]\!]$ so that the corresponding Pfaffians generate the vector space spanned by all Pfaffians, leading to a complexity of

$$\mathcal{O}\left(\binom{K+1}{2}^{\omega}\right) \tag{32}$$

which is significantly less. It is unclear however how we can choose these subsets and ensure that the Pfaffians defined by these subsets are enough. In practice, we use another heuristic to avoid computing all the Pfaffians that is given in Appendix B. We give below the computation time it took us in practice to compute the first Macaulay matrix (Step 3 of Algorithm 2) and echelonize it, for $r = 5$ and several values of m, and with $n = 2^m$.

m	K	$\binom{rm-s}{4}$	# Pfaffians	T_{write} (s)	T_{Gauss} (s)
8	190	23751	18000	2.8	1.1
9	201	46376	27000	3.3	2.1
10	216	82251	42000	6.58	5.2
11	231	135751	63000	8.2	9.1
12	246	211876	77000	15.7	17.1
13	261	316251	105000	27.7	29.9

Table 5: Time T_{write} to compute the Pfaffians in seconds and time T_{Gauss} (in seconds) to echelonize the resulting Macaulay matrix, for full-length dual Goppa codes of degree $r = 5$. We also report the total number of Pfaffians *i.e.* $\binom{rm-s}{4}$, and the number of Pfaffians that we computed in practice (# Pfaffians) – essentially those corresponding to Heuristic 2 –. Each time, the Pfaffians that we computed were enough to solve the system.

We see that the number of variables K in the Pfaffian modeling is asymptotically proportional to m, so we expect the Gaussian elimination time to be proportional to a power of m. The pessimistic complexity (31) suggests that the exponent is less than $4\omega \leq 12$, while the theoretical optimum (32) suggests it should be around 2ω. In practice, a simple linear regression between m and T_{Gauss} gives an exponent of 6.8945 with confidence $R^2 = 0.9973$. This is significantly higher than 2ω, but this might be explained by the fact that we compute significantly more Pfaffians than $\binom{K+1}{2}$, leading to a Macaulay matrix with many more rows than columns. The reason why we do this instead of computing around $\binom{K+1}{2}$ random Pfaffians (which would indeed be enough) is that the first Pfaffians that we compute, *i.e.* those outside the dense block, are sparse. We have seen in practice that computing random Pfaffians increases T_{write} so much that our strategy is the fastest for solving the TII instances reported on Table 1. For the CFS parameters on Table 2 however, choosing random Pfaffians might be a winning strategy due to the very large number of rows given by our heuristic.

5.3 Finishing the attack with enough rank 2 matrices in $\mathscr{C}_{\mathbf{mat}}$

Computing a fundamental triple $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w})$ *of vectors.* Rank 2 matrices in the matrix code of relations of the shortened codes $\mathscr{C}_s^i$ we immediately deduce rank 2 matrices in the matrix code of relations of the original code. For each such matrix $\boldsymbol{B}$ of rank 2, we compute a pair $(\boldsymbol{\mu}, \boldsymbol{\nu})$ of vectors that span the rowspace of $\boldsymbol{B}$. We are going to deduce from this a triple $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w})$ of elements of $\mathscr{C}$ which live in a much smaller space related to the GRS code we want to retrieve: they will live namely in $\bigoplus_{j=0}^{e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$ (or a Frobenius conjugate of this code). Here is how we proceed. We first compute a basis $\mathcal{B}$ of $\mathscr{C}$ which is of the form

$$\mathcal{B} = (\boldsymbol{b}_0, \ldots, \boldsymbol{b}_{r-1}, \boldsymbol{b}_0^2, \ldots, \boldsymbol{b}_{r-1}^2, \ldots, \boldsymbol{b}_0^{2^{m-1}}, \ldots, \boldsymbol{b}_{r-1}^{2^{m-1}}), \tag{33}$$

and define $\boldsymbol{H}_{\mathcal{B}}$ as the matrix whose rows are the elements of $\mathcal{B}$ in that order. We then define

$$\boldsymbol{u} \stackrel{\text{def}}{=} \boldsymbol{\mu}\boldsymbol{H}_{\mathcal{B}} \in \mathscr{C}$$
$$\boldsymbol{v} \stackrel{\text{def}}{=} \boldsymbol{\nu}\boldsymbol{H}_{\mathcal{B}} \in \mathscr{C}.$$

We also compute a vector $\boldsymbol{w} \in \mathscr{C}$ such that

$$\boldsymbol{u} \star \boldsymbol{v} = \boldsymbol{w}^2.$$

In other words $\boldsymbol{w} = (\boldsymbol{u} \star \boldsymbol{v})^{\frac{1}{2}}$. It turns that these vectors live in a much smaller space than the whole code $\mathscr{C}$ as explained above. We observed this experimentally and proved it for $\boldsymbol{u}$ and $\boldsymbol{v}$ under some mild assumption

Proposition 5.3. *Under Assumption 3, we have for some $a \in [\![0, m-1]\!]$:*

$$\boldsymbol{u}, \boldsymbol{v} \in \bigoplus_{j=a}^{a+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right).$$

For a proof see Appendix C. We experimentally also find that $\boldsymbol{w}$ lives in the same space as $\boldsymbol{u}$ and $\boldsymbol{v}$. From now on we call FundamentalTriple($\boldsymbol{B}$) the function that outputs such a triple $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w})$.

Alignment of Fundamental triples. With the previous procedure we obtain many triples $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)$ which live in subspaces of the form $\bigoplus_{j=a_i}^{a_i+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$. We say that two triples $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)$ and $(\boldsymbol{u}_j, \boldsymbol{v}_j, \boldsymbol{w}_j)$ are *aligned* when $a_i = a_j$. Notice that there exists necessarily for two such triples an integer $b \in [\![0, m-1]\!]$ such that $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)$ and $(\boldsymbol{u}_j^{2^b}, \boldsymbol{v}_j^{2^b}, \boldsymbol{w}_j^{2^b})$ are aligned. This integer b can be found with the following heuristic

Heuristic 1. *Let $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w})$ and $(\boldsymbol{u}', \boldsymbol{v}', \boldsymbol{w}')$ be two triples obtained as explained above, each of them living respectively in $\bigoplus_{j=a}^{a+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$ and in $\bigoplus_{j=a'}^{a'+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$. For $b \in [\![0, m-1]\!]$ define the linear code $\mathscr{V}_b \stackrel{\text{def}}{=} \left\langle \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}, \boldsymbol{u}'^{2^b}, \boldsymbol{v}'^{2^b}, \boldsymbol{w}'^{2^b} \right\rangle_{\mathbb{F}_{2^m}}$. We assume that $\dim \mathscr{V}_i = 6$ all the time, otherwise we can merely check this dimension to verify alignment. Finally, define the function $f : \{0, \ldots, m-1\} \to \mathbb{N}$ as $f(b) \stackrel{\text{def}}{=} \dim \mathscr{V}_b^{\star 2}$. Then*

- *f is not a constant function;*
- *f is minimal at b_0 such that $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i)$ and $(\boldsymbol{u}_j^{2^{b_0}}, \boldsymbol{v}_j^{2^{b_0}}, \boldsymbol{w}_j^{2^{b_0}})$ are aligned;*
- *f is maximal at $b_0 + \lfloor m/2 \rfloor \mod m$.*

We observed this experimentally and some justification is given in the appendix. From now on we denote by Alignement($(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}), (\boldsymbol{u}', \boldsymbol{v}', \boldsymbol{w}')$) the function which outputs such an integer b_0.

Recovering a GRS code. By finding $\lceil e_{\mathscr{G}}/3 \rceil$ triples we can reconstruct a large subcode of (some conjugate of) $\mathscr{D} \stackrel{\text{def}}{=} \bigoplus_{j=0}^{e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$. We may even assume that the triples generate this code entirely, up to conjugation. In fact, the only parameter for which we could not reconstruct this code entirely was $r = 5$, for which Assumption 3 is refined. We explain in Appendix E how to recover a support and Goppa polynomial in this case anyway, leading to a correct attack on TII challenges 83 and 89.

From now on we assume that $\mathscr{D}$ is reconstructed. We are going to recover (up to Frobenius conjugation) the GRS code $\mathbf{GRS}_r\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right)$ by using the following proposition.

Proposition 5.4. *Define* $\mathscr{D}_0 \stackrel{\text{def}}{=} \mathscr{D}$ *and* $\mathscr{D}_{i+1} \stackrel{\text{def}}{=} \mathscr{D}_i \cap \mathscr{D}_i^{(2)}$ *for all* $0 \leq i < e_{\mathscr{G}} - 1$. *Then we have*

$$\forall 0 \leq i < e_{\mathscr{G}},\ \mathscr{D}_i = \bigoplus_{j=i}^{e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right).$$

Proof. We proceed through induction. The desired equality is true for $i = 0$ by definition. Let us assume it is true for some $0 \leq i < e_{\mathscr{G}} - 1$, and let $\boldsymbol{u} \in \mathscr{D}_i \cap \mathscr{D}_i^{(2)}$. We may write $\boldsymbol{u} = \sum_{j=i}^{e_{\mathscr{G}}-1} \boldsymbol{y}^{2^j} f_j(\boldsymbol{x})^{2^j}$, with polynomials $f_j \in \mathbb{F}_{2^m}[X]_{<r}$ being unique. On the other hand, we know that $\boldsymbol{u} \in \mathscr{D}_i^{(2)}$, which means that there are unique polynomials g_j such that

$$\boldsymbol{u} = \left(\sum_{j=i}^{e_{\mathscr{G}}-1} \boldsymbol{y}^{2^j} g_j(\boldsymbol{x})^{2^j}\right)^2 = \sum_{j=i+1}^{e_{\mathscr{G}}} \boldsymbol{y}^{2^j} g_{j-1}(\boldsymbol{x})^{2^j}.$$

The uniqueness of the writing of $\boldsymbol{u}$ as an element of $\mathscr{C}_{\mathbb{F}_{2^m}} = \bigoplus_{j=0}^{m-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$ implies that we have $f_j = g_{j-1}$ for all $j \geq i+1$ and $f_i = 0$, hence we have indeed $\boldsymbol{u} \in \bigoplus_{j=i+1}^{e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$. The converse inclusion is easily verified. □

In particular, we get access to $\mathscr{D}_{e_{\mathscr{G}}-1} = \mathbf{GRS}_r\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right)$. It only remains to apply the Sidelnikov-Shestakov attack to recover a valid pair $\boldsymbol{x}', \boldsymbol{y}'$ such that $\mathscr{C} = \mathscr{A}_r(\boldsymbol{x}', \boldsymbol{y}')^{\perp}$. We obtain a Goppa polynomial Γ' by using the method described in [Mor25, §3]. The whole attack is then given by the following algorithm.

The complexity of Algorithm 3 is that of linear algebra over matrices of size $rm \times rm$, repeated at most $mt = \mathcal{O}(m \log r)$ times. This is by far negligible compared to the complexity of Algorithm 2. We conclude

Theorem 5.5. *Under our heuristic experimentally supported, we retrieve the private structure of a square distinguishable binary Goppa code of degree* r *and extension degree* m *by running* $t = \lceil e_{\mathscr{G}}/3 \rceil$ *times Algorithm 2 and then Algorithm 3, for a total complexity of*

$$\kappa_{tot} = \mathcal{O}\left(\binom{rm - s_{\mathscr{G}} + 1}{4}^{\omega} \log r\right). \tag{34}$$

Algorithm 3 Recovering a support and a Goppa polynomial from $t > e_{\mathscr{G}}/3$ matrices of rank 2 in $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ where $\mathcal{B}$ is a given basis of $\mathscr{C}$.

Input: $\boldsymbol{B}_1, \ldots, \boldsymbol{B}_t$ of rank 2 in $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ where $\mathcal{B}$ is a basis of $\mathscr{C}$ in the form of (33).
Output: $\boldsymbol{x}', \Gamma'$ such that $\mathscr{C} = \mathscr{G}(\boldsymbol{x}', \Gamma')^{\perp}$.
$(\boldsymbol{u}_1, \boldsymbol{v}_1, \boldsymbol{w}_1) \leftarrow \textsc{FundamentalTriple}(\boldsymbol{B}_1)$;
for $i = 2, \ldots, t$ **do**
    $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i) \leftarrow \textsc{FundamentalTriple}(\boldsymbol{B}_i)$;
    $b \leftarrow \textsc{Alignement}((\boldsymbol{u}_1, \boldsymbol{v}_1, \boldsymbol{w}_1), \boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i))$;
    $(\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i) \leftarrow (\boldsymbol{u}_i^{2^b}, \boldsymbol{v}_i^{2^b}, \boldsymbol{w}_i^{2^b})$;
$\mathscr{D} \leftarrow \langle \boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i \mid 1 \leq i \leq t \rangle_{\mathbb{F}_{2^m}}$;
for $i = 1, \ldots, e_{\mathscr{G}} - 1$ **do**
    $\mathscr{D} \leftarrow \mathscr{D} \cap \mathscr{D}^{(2)}$;
Use [Mor25, §3] to recover $\boldsymbol{x}', \Gamma'$ from $\mathscr{D} = \mathbf{GRS}_r\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right)$ (up to conjugation).

6 Concluding Remarks

With the notable exception of the very peculiar case when the Goppa polynomial is of degree 2 [Mor25], this is the first time that there is a key attack on the McEliece scheme based on binary Goppa codes. It applies to square-distinguishable binary Goppa codes, meaning very high rate Goppa codes. It gives in this case an attack of polynomial complexity. In particular it applies to the CFS signature scheme [CFS01]. It breaks this 25 years old signature scheme. We also applied it to codes of smaller rate which are not square-distinguishable such as the TII challenges we broke. Arguably however, the codes of this kind were close to be square-distinguishable.

This shows that the MinRank modeling with Pfaffians does not only lead to distinguisher attacks, it may also lead to actual attacks. We prove in this case some fundamental results about what happens in this case when we shorten the dual code before applying our attack to it and give in the alternant case a tight bound about the number of positions that we can shorten and still keep rank 2 matrices in the matrix code of quadratic relationships. Moreover we also observed that there are squares in the associated polynomial at degree 2. They can easily be obtained by solving a linear system and are instrumental for keeping the computations of the Gröbner basis at degree 2 in this case. Here we have only some partial understanding of the dimension of the vector space formed by these squares (and this particularly if we shorten the code). In any case, the Pfaffian modeling of [CMT23b] raises very intriguing questions on understanding the subspace of squares which are in the corresponding ideal. It suggests that there are many interesting algebraic phenomena in the Pfaffian modeling which would be interesting to understand in order to push further the attack proposed here way beyond the square-distinguishability regime.

References

ABC+22. Martin Albrecht, Daniel J. Bernstein, Tung Chou, Carlos Cid, Jan Gilcher, Tanja Lange, Varun Maram, Ingo von Maurich, Rafael Mizoczki, Ruben Niederhagen, Edoardo Persichetti, Kenneth Paterson, Christiane Peters, Peter Schwabe, Nicolas Sendrier, Jakub Szefer, Cen Jung Tjhai, Martin Tomlinson, and Wang Wen. Classic McEliece (merger of Classic McEliece and NTS-KEM). https://classic.mceliece.org, November 2022. Fourth round finalist of the NIST post-quantum cryptography call.

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

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

BCGO09. Thierry P. Berger, Pierre-Louis Cayrel, Philippe Gaborit, and Ayoub Otmani. Reducing key length of the McEliece cryptosystem. In Bart Preneel, editor, *Progress in Cryptology - AFRICACRYPT 2009*, volume 5580 of *LNCS*, pages 77–97, Gammarth, Tunisia, June 21-25 2009.

Ber10. Daniel J. Bernstein. Grover vs. McEliece. In Nicolas Sendrier, editor, *Post-Quantum Cryptography 2010*, volume 6061 of *LNCS*, pages 73–80. Springer, 2010.

BJMM12. Anja Becker, Antoine Joux, Alexander May, and Alexander Meurer. Decoding random binary linear codes in $2^{n/20}$: How $1 + 1 = 0$ improves information set decoding. In *Advances in Cryptology - EUROCRYPT 2012*, LNCS. Springer, 2012.

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

BLM11. Paulo Barreto, Richard Lindner, and Rafael Misoczki. Monoidic codes in cryptography. In *Post-Quantum Cryptography 2011*, volume 7071 of *LNCS*, pages 179–199. Springer, 2011.

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

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

BLP11. Daniel J. Bernstein, Tanja Lange, and Christiane Peters. Wild McEliece Incognito. In Bo-Yin Yang, editor, *Post-Quantum Cryptography 2011*, volume 7071 of *LNCS*, pages 244–254. Springer Berlin Heidelberg, 2011.

BM17. Leif Both and Alexander May. Optimizing BJMM with Nearest Neighbors: Full Decoding in $2^{2/21n}$ and McEliece Security. In *WCC Workshop on Coding and Cryptography*, September 2017.

BMT24. Magali Bardet, Rocco Mora, and Jean-Pierre Tillich. Polynomial time key-recovery attack on high rate random alternant codes. *IEEE Trans. Inform. Theory*, 70(6):4492–4511, 2024.

BMvT78. Elwyn Berlekamp, Robert McEliece, and Henk van Tilborg. On the inherent intractability of certain coding problems. *IEEE Trans. Inform. Theory*, 24(3):384–386, May 1978.

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

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

CCMZ15. Igniacio Cascudo, Ronald Cramer, Diego Mirandola, and Gilles Zémor. Squares of random linear codes. *IEEE Trans. Inform. Theory*, 61(3):1159–1173, 3 2015.

CDMT22. Kevin Carrier, Thomas Debris-Alazard, Charles Meyer-Hilfiger, and Jean-Pierre Tillich. Statistical decoding 2.0: Reducing decoding to LPN. In *Advances in Cryptology - ASIACRYPT 2022*, LNCS. Springer, 2022.

CDMT24. Kévin Carrier, Thomas Debris-Alazard, Charles Meyer-Hilfiger, and Jean-Pierre Tillich. Reduction from sparse LPN to LPN, dual attack 3.0. In Marc Joye and Gregor Leander, editors, *Advances in Cryptology - EUROCRYPT 2024 - 43rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zurich, Switzerland, May 26-30, 2024, Proceedings, Part VI*, volume 14656 of *LNCS*, pages 286–315. Springer, 2024. Artifact available at https://artifacts.iacr.org/eurocrypt/2024/a10/.

CFS01. Nicolas Courtois, Matthieu Finiasz, and Nicolas Sendrier. How to achieve a McEliece-based digital signature scheme. In *Advances in Cryptology - ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 157–174, Gold Coast, Australia, 2001. Springer.

CGG+14. Alain Couvreur, Philippe Gaborit, Valérie Gauthier-Umaña, Ayoub Otmani, and Jean-Pierre Tillich. Distinguisher-based attacks on public-key cryptosystems using Reed-Solomon codes. *Des. Codes Cryptogr.*, 73(2):641–666, 2014.

CMT23a. Alain Couvreur, Rocco Mora, and Jean-Pierre Tillich. A new approach based on quadratic forms to attack the McEliece cryptosystem. In Jian Guo and Ron Steinfeld, editors, *Advances in Cryptology - ASIACRYPT 2023 - 29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, China, December 4-8, 2023, Proceedings, Part IV*, volume 14441 of *LNCS*, pages 3–38. Springer, 2023.

CMT23b. Alain Couvreur, Rocco Mora, and Jean-Pierre Tillich. A new approach based on quadratic forms to attack the McEliece cryptosystem. *arXiv preprint arXiv:2306.10294*, 2023.

COT14. Alain Couvreur, Ayoub Otmani, and Jean-Pierre Tillich. New identities relating wild Goppa codes. *Finite Fields Appl.*, 29:178–197, 2014.

DEEK24. Léo Ducas, Andre Esser, Simona Etinski, and Elena Kirshanova. Asymptotics and improvements of sieving for codes. In Marc Joye and Gregor

Leander, editors, *Advances in Cryptology - EUROCRYPT 2024 - 43rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zurich, Switzerland, May 26-30, 2024, Proceedings, Part VI*, volume 14656 of *Lecture Notes in Computer Science*, pages 151–180. Springer, 2024.

DMR11. Hang Dinh, Cris Moore, and Alexander Russell. The McEliece cryptosystem resists quantum Fourier sampling attacks. In *Advances in Cryptology - CRYPTO 2011*, volume 6841 of *LNCS*, pages 761–779. Springer, 2011.

Dum89. Il'ya Dumer. Two decoding algorithms for linear codes. *Probl. Inf. Transm.*, 25(1):17–23, 1989.

Ess22. Andre Esser. Revisiting nearest-neighbor-based information set decoding. Cryptology ePrint Archive, Paper 2022/1328, 2022. https://eprint.iacr.org/2022/1328.

EZ23. Andre Esser and Floyd Zweydinger. New time-memory trade-offs for subset sum - improving ISD in theory and practice. In Carmit Hazay and Martijn Stam, editors, *Advances in Cryptology - EUROCRYPT 2023 - 42nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Lyon, France, April 23-27, 2023, Proceedings, Part V*, volume 14008 of *Lecture Notes in Computer Science*, pages 360–390. Springer, 2023.

FGLM93. Jean-Charles Faugère, Patrizia M. Gianni, Daniel Lazard, and Teo Mora. Efficient computation of zero-dimensional gröbner bases by change of ordering. *J. Symbolic Comput.*, 16(4):329–344, 1993.

FGO$^+$11. Jean-Charles Faugère, Valérie Gauthier, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. A distinguisher for high rate McEliece cryptosystems. In *Proc. IEEE Inf. Theory Workshop- ITW 2011*, pages 282–286, Paraty, Brasil, October 2011.

FGO$^+$13. Jean-Charles Faugère, Valérie Gauthier, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. A distinguisher for high rate McEliece cryptosystems. *IEEE Trans. Inform. Theory*, 59(10):6830–6844, October 2013.

Fin03. Matthieu Finiasz. Words of minimal weight and weight distribution of binary Goppa codes. In *Proceedings of the 2003 IEEE International Symposium on Information Theory*. IEEE, 2003.

FOPT10. Jean-Charles Faugère, Ayoub Otmani, Ludovic Perret, and Jean-Pierre Tillich. Algebraic cryptanalysis of McEliece variants with compact keys. In *Advances in Cryptology - EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 279–298, 2010.

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

Gib91. J. K. Gibson. Equivalent Goppa codes and trapdoors to McEliece's public key cryptosystem. In Donald Davies, editor, *Advances in Cryptology - EUROCRYPT 1991*, volume 547 of *LNCS*, pages 517–521. Springer Berlin / Heidelberg, 1991.

KT17. Ghazal Kachigar and Jean-Pierre Tillich. Quantum information set decoding algorithms. In *Post-Quantum Cryptography 2017*, volume 10346 of *LNCS*, pages 69–89, Utrecht, The Netherlands, June 2017. Springer.

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

Lem25. Axel Lemoine. The tangent space attack. Cryptology ePrint Archive, Paper 2025/763, 2025.

LL97. Françoise Levy-dit-Vehel and Simon Litsyn. Parameters of Goppa codes revisited. *IEEE Trans. Inform. Theory*, 43(6):1811–1819, November 1997.

LMT25. Axel Lemoine, Rocco Mora, and Jean-Pierre Tillich. Understanding the new distinguisher of alternant codes at degree 2. *IACR Cryptol. ePrint Arch.*, page 531, 2025.

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

McE78. Robert J. McEliece. *A Public-Key System Based on Algebraic Coding Theory*, pages 114–116. Jet Propulsion Lab, 1978. DSN Progress Report 42-44.

MMT11. Alexander May, Alexander Meurer, and Enrico Thomae. Decoding random linear codes in $O(2^{0.054n})$. In Dong Hoon Lee and Xiaoyun Wang, editors, *Advances in Cryptology - ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 107–124. Springer, 2011.

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

Mor25. Rocco Mora. On the matrix code of quadratic relationships for a goppa code. *Adv. Math. Commun.*, 19(3):829–852, 2025.

MP12. Irene Márquez-Corbella and Ruud Pellikaan. Error-correcting pairs for a public-key cryptosystem, 2012.

MS77. F. Jessie MacWilliams and N. J. A. Sloane. *The Theory of Error-Correcting Codes*. North-Holland, Amsterdam, 1977.

MS86. Florence J. MacWilliams and Neil J. A. Sloane. *The Theory of Error-Correcting Codes*. North–Holland, Amsterdam, fifth edition, 1986.

MT22. Rocco Mora and Jean-Pierre Tillich. On the dimension and structure of the square of the dual of a Goppa code. In *WCC 2022 - Workshop on Coding Theory and Cryptography*, 2022.

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

Pan25. Lorenz Panny. On breaking McEliece keys using brute force. Cryptology ePrint Archive, Paper 2025/632, 2025.

Ran25. Hugues Randriambololona. The syzygy distinguisher. In Serge Fehr and Pierre-Alain Fouque, editors, *Advances in Cryptology - EUROCRYPT 2025 - 44th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Madrid, Spain, May 4-8, 2025, Proceedings, Part VI*, volume 15606 of *Lecture Notes in Computer Science*, pages 324–354. Springer, 2025.

RSA78. Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. A method for obtaining digital signatures and public-key cryptosystems. *Commun. ACM*, 21(2):120–126, 1978.

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

SS92. Vladimir Michilovich Sidelnikov and S.O. Shestakov. On the insecurity of cryptosystems based on generalized Reed-Solomon codes. *Discrete Math. Appl.*, 1(4):439–444, 1992.

Ste88. Jacques Stern. A method for finding codewords of small weight. In G. D. Cohen and J. Wolfmann, editors, *Coding Theory and Applications*, volume 388 of *LNCS*, pages 106–113. Springer, 1988.

Tra24. Bénédikt Minh Dang Tran. *Post-Quantum Code-Based Cryptography.* PhD thesis, EPFL, Lausanne, 2024.

Wie10. Christian Wieschebrink. Cryptanalysis of the Niederreiter public key scheme based on GRS subcodes. In *Post-Quantum Cryptography 2010*, volume 6061 of *LNCS*, pages 61–72. Springer, 2010.

Wie25. Andreas Wiemers. A note on the Goppa code distinguishing problem. Cryptology ePrint Archive, Paper 2025/1661, 2025.

A Proofs of the Lemmas needed in §3

We are going to prove here lemmas used to prove Theorem 3.7. We first need a result which is also proved in [Ran25], namely that (see the proof of Proposition 4 in [Ran25])

Lemma A.1. *Let* $\boldsymbol{p}_j = (\underbrace{0,\cdots,0}_{k-j\ terms},1)^\intercal \in \mathbb{F}_{q^m}^{k-j+1}$. *We consider the evaluation map* $\boldsymbol{e}_j : \mathcal{R}_{j-1} \mapsto \mathbb{F}_{q^m}$ *at* $\boldsymbol{p}_j$ *which maps* f *to its evaluation* $f(\boldsymbol{p}_j)$. *It extends naturally coordinatewise as* $\boldsymbol{e}_j : \mathcal{R}_{j-1}^2 \to \mathbb{F}_{q^m}^2$. *We have for any* $j \in [\![1, s]\!]$

$$\dim\left(\boldsymbol{e}_j(V_{j-1})\right) \leq 1.$$

We give a proof for the sake of self-containedness, although a proof of this statement is given in the proof of [Ran25, Prop. 4].

Proof. By an immediate proof by induction we have for any $j \in [\![1, s]\!]$ that $V_j = V_{j-1} \cap \mathcal{R}_{j-1}^2$ and

$$V_j = \ker(\boldsymbol{e}_j : V_{j-1} \to \mathbb{F}_{q^m}^2). \tag{35}$$

Since the 2×2 minors of $\boldsymbol{\Phi}_{j-1}$ are in $I_2(\mathscr{C}_{j-1})$, they all vanish when evaluated on $\boldsymbol{c}_1, \cdots, \boldsymbol{c}_{k-j+1}$. In particular the last coordinate of the corresponding vector is equal to 0. This coordinate corresponds to the last column of $\boldsymbol{G}_{j-1}$, which is nothing but $\boldsymbol{p}_j$. This shows that such minors vanish when evaluated on $\boldsymbol{p}_j$. This immediately implies that $\dim\left(\boldsymbol{e}_j(V_{j-1})\right) \leq 1$. □

From this it immediately follows the lemma we gave in §3

Lemma 3.2. *We have*

$$\dim V_j \geq \dim V_{j-1} - 1,\ \forall j \in [\![1, s]\!] \tag{25}$$
$$\dim V_j \geq \dim V_{\boldsymbol{\Phi}} - j \tag{26}$$

where $V_0 \stackrel{\text{def}}{=} V_{\boldsymbol{\Phi}}$. *If we let* f_j *be the dimension of* V_j *for any* $j \in [\![1, s]\!]$ *then we have that* $f_j \geq f_{\mathscr{A}} - j$, *where* $f_{\mathscr{A}} \stackrel{\text{def}}{=} (e_{\mathscr{A}} + 1)r - \frac{q^{e_{\mathscr{A}}+1}-1}{q-1}$ *with* $e_{\mathscr{A}} \stackrel{\text{def}}{=} \lfloor \log_q(r-1) \rfloor$.

Proof. For any j in $[\![1, s]\!]$, we have

$$\dim \boldsymbol{e}_j(V_{j-1}) + \dim \ker(\boldsymbol{e}_j : V_{j-1} \to \mathbb{F}_{q^m}^2) = \dim V_{j-1}.$$

Since on one hand, $\dim\left(\boldsymbol{e}_j(V_{j-1})\right) \leq 1$ by Lemma A.1 and $\dim \ker(\boldsymbol{e}_j : V_{j-1} \to \mathbb{F}_{q^m}^2) = \dim V_j$ by (35), we deduce that for any j in $[\![1, s]\!]$

$$\dim V_j \geq \dim V_{j-1} - 1.$$

The remaining statements follow directly from this and from the fact that $V_0 = \dim V_{\boldsymbol{\Phi}} = f_{\mathscr{A}}$. The last equality is a direct consequence that all the entries of the first row of $\boldsymbol{\Phi}$ involve different variables. This shows that the dimension of $V_{\boldsymbol{\Phi}}$ is nothing but the number of columns of $\boldsymbol{\Phi}$ which is readily seen to be equal to $f_{\mathscr{A}}$. □

Let us prove now the crucial Lemma 3.3 that we recall here

Lemma 3.3. *Let* $\tilde{e}_{\mathscr{A}} \stackrel{\text{def}}{=} \lfloor \log_q \frac{r}{2} \rfloor$ *and* $s_{\mathscr{A}} \stackrel{\text{def}}{=} (\tilde{e}_{\mathscr{A}}+1)r - 2\frac{q^{\tilde{e}_{\mathscr{A}}+1}-1}{q-1}$. *We have for any* $j \in [\![0, s_{\mathscr{A}}]\!]$,

$$\dim\left(V_j^1 \cap V_j^2\right) \geq s_{\mathscr{A}} - j.$$

To prove this lemma we first prove that it is is true for $j = 0$.

Lemma A.2.

$$\dim(V_0^1 \cap V_0^2) = s_{\mathscr{A}}.$$

Proof. The dimensions of the subspaces we are interested in do not depend on the chosen generator matrix. It is convenient here to consider the canonical basis (15). In this basis the matrix $\mathbf{\Phi}$ is given by

$$\boldsymbol{B}_u \stackrel{\text{def}}{=} \begin{pmatrix} X_0^{(0)} & X_1^{(0)} & \cdots & X_{r-1-q^u}^{(0)} \\ X_{q^u}^{(0)} & X_{q^u+1}^{(0)} & \cdots & X_{r-1}^{(0)} \end{pmatrix}$$
$$\mathbf{\Phi} \stackrel{\text{def}}{=} \left(\boldsymbol{B}_0^{(e)} \middle| \boldsymbol{B}_1^{(e-1)} \middle| \cdots \middle| \boldsymbol{B}_e^{(0)} \right)$$

where $e = e_{\mathscr{A}}$ (as defined in (10)).

From this form it is readily seen that $\dim(V_0^1 \cap V_0^2)$ is given by the number N of variables that appear both in the first row and the second row of $\mathbf{\Phi}$. We clearly have

$$\begin{aligned} \dim(V_0^1 \cap V_0^2) = N &= \sum_{u=0}^{e} \max\left(r - 2q^u, 0\right) \\ &= \sum_{u=0}^{\tilde{e}_{\mathscr{A}}} (r - 2q^u) \\ &= (\tilde{e}_{\mathscr{A}}+1)r - 2\frac{q^{\tilde{e}_{\mathscr{A}}+1}-1}{q-1}. \end{aligned} \tag{36}$$

□

Lemma 3.3 is clearly a direct consequence of this lemma and the induction inequality given below.

Lemma A.3. *For all* $j \in [\![1, s_{\mathscr{A}}]\!]$ *we have*

$$\dim\left(V_j^1 \cap V_j^2\right) \geq \dim\left(V_{j-1}^1 \cap V_{j-1}^2\right) - 1. \tag{37}$$

Proof. To verify this lemma, let us use Lemma A.1 which says that $\dim\left(\boldsymbol{e}_j(V_{j-1})\right) \leq 1$. This implies that one the following points holds

(i) there exists an element $\lambda \in \mathbb{F}_{q^m}^*$ such that $\boldsymbol{e}_j(V_{j-1}) = \left\{ \begin{pmatrix} x \\ \lambda x \end{pmatrix}, x \in \mathbb{F}_{q^m} \right\}$ or

(ii) $\boldsymbol{e}_j(V_{j-1}) = \left\{ \begin{pmatrix} x \\ 0 \end{pmatrix}, x \in \mathbb{F}_{q^m} \right\}$ or

(iii) $\boldsymbol{e}_j(V_{j-1}) = \left\{ \begin{pmatrix} 0 \\ x \end{pmatrix}, x \in \mathbb{F}_{q^m} \right\}$ or

(iv) $\boldsymbol{e}_j(V_{j-1}) = \{0\}$.

Case (i): Let us assume that (i) holds. Consider now the restriction of $\boldsymbol{e}_j$ to $V_{j-1}^1 \cap V_{j-1}^2$. Let

$$W \stackrel{\text{def}}{=} \ker(\boldsymbol{e}_j : V_{j-1}^1 \cap V_{j-1}^2 \to \mathbb{F}_{q^m}).$$

Let $w \in W$. By definition of W, all three following properties hold

(a) $\boldsymbol{e}_j(w) = 0$.

(b) there exists v such that $\begin{pmatrix} w \\ v \end{pmatrix}$ is in V_{j-1}

(c) there exists x such that $\begin{pmatrix} x \\ w \end{pmatrix}$ is in V_{j-1}

From (i) and (a) $\boldsymbol{e}_j(w) = 0$, we know that we have at the same time $\boldsymbol{e}_j(v) = \boldsymbol{e}_j(x) = 0$. This implies that $\begin{pmatrix} w \\ v \end{pmatrix}$ and $\begin{pmatrix} x \\ w \end{pmatrix}$ are both in V_j, which in turn implies that w is in V_j^1 and V_j^2. In other words, we have proved that $W \subseteq V_j^1 \cap V_j^2$ and since $\dim W \geq \dim \left(V_{j-1}^1 \cap V_{j-1}^2\right) - 1$, this shows that (37) holds.

Case (ii): Consider that (ii) holds. It will be helpful to consider for an element v_2 in $\pi_2(V_{j-1})$ the affine set

$$\mathcal{A}(v_2) = \pi_1\left(\pi_2^{-1}(v_2)\right) = \left\{ x \in \mathcal{R}_{j-1} : \begin{pmatrix} x \\ v_2 \end{pmatrix} \in V_{j-1} \right\}.$$

If we let $W = \mathcal{A}(0) = \left\{ x \in \mathcal{R}_{j-1} : \begin{pmatrix} x \\ 0 \end{pmatrix} \in V_{j-1} \right\}$, then $\mathcal{A}(v_2)$ is a coset of W. There are two cases to consider.

Subcase $\boldsymbol{e}_j(W) = \mathbb{F}_{q^m}$. In this case, we claim that $V_j^1 \cap V_j^2 = V_{j-1}^1 \cap V_{j-1}^2$. Indeed let w be in $V_{j-1}^1 \cap V_{j-1}^2$. There exist in this case v and x such that both $\begin{pmatrix} w \\ v \end{pmatrix}$ and $\begin{pmatrix} x \\ w \end{pmatrix}$ belong to V_{j-1}. From the form of $\boldsymbol{e}_j(V_{j-1})$ and the fact that $\begin{pmatrix} x \\ w \end{pmatrix}$ belongs to V_{j-1} this implies that $\boldsymbol{e}_j(w) = 0$. Since $\boldsymbol{e}_j(v) = 0$ because v is V_j^2, this implies that $\begin{pmatrix} w \\ v \end{pmatrix}$ is in V_j and therefore w is in V_j^1. From the fact that $\boldsymbol{e}_j(W) = \mathbb{F}_{q^m}$, there exists y in W such that $\boldsymbol{e}_j(y) = \boldsymbol{e}_j(x)$. Since $\begin{pmatrix} x - y \\ w \end{pmatrix}$ also belongs to V_{j-1} by definition of W and since $\boldsymbol{e}_j(x - y) = 0$ we also have that w belongs to V_j^2. This implies $w \in V_j^1 \cap V_j^2$ and shows that $V_j^1 \cap V_j^2 = V_{j-1}^1 \cap V_{j-1}^2$.

Subcase $\boldsymbol{e}_j(W) = \{0\}$. Consider the linear map

$$\phi : \pi_2(V_{j-1}) \to \mathcal{R}_{j-1}/W$$

which maps an element $u \in \pi_2(V_{j-1})$ to the coset $u' + W$ which is such that $(u'+W) \times \{u\} \subseteq V_{j-1}$. Note that $\boldsymbol{e}_j$ is well defined on $\mathcal{R}_{j-1}/W$ because $\boldsymbol{e}_j(W) =$

$\{0\}$. Let $L : \pi_2(V_{j-1}) \to \mathbb{F}_{q^m}$ be the linear map $L = \boldsymbol{e}_j \circ \phi$. Consider the kernel of its restriction to $V^1_{j-1} \cap V^2_{j-1}$, $U \stackrel{\text{def}}{=} \ker(L : V^1_{j-1} \cap V^2_{j-1} \to \mathbb{F}_{q^m})$. We have

$$\dim U \geq \dim \left(V^1_{j-1} \cap V^2_{j-1}\right) - 1. \tag{38}$$

Let u be in U. Since $U \subseteq V^1_{j-1} \cap V^2_{j-1}$ there exist v and x such that both $\begin{pmatrix} u \\ v \end{pmatrix}$ and $\begin{pmatrix} x \\ u \end{pmatrix}$ belong to V_{j-1}. From the form of $\boldsymbol{e}_j(V_{j-1})$ and the fact that $\begin{pmatrix} x \\ u \end{pmatrix}$ belongs to V_{j-1} this implies that $\boldsymbol{e}_j(u) = 0$. From this, we deduce that $\begin{pmatrix} u \\ v \end{pmatrix}$ is in V_j and that $u \in V^1_j$. Moreover by definition of U, we also have $\boldsymbol{e}_j(x) = 0$. From this we deduce that $\begin{pmatrix} x \\ u \end{pmatrix}$ is in V_j and therefore u is also in V^2_j. Putting both facts together, we get that $u \in V^1_j \cap V^2_j$. In other words, $U \subseteq V^1_j \cap V^2_j$. This together with (38) shows (37).
Case (iii): (37) is proved similarly to Case (ii).
Case (iv): In this case $V_j = V_{j-1}$ and we immediately deduce that

$$\dim \left(V^1_j \cap V^2_j\right) = \dim \left(V^1_{j-1} \cap V^2_{j-1}\right).$$

□

Let us recall Lemma 3.4 that we also prove here

Lemma 3.4. *We have for any $s \in [\![0, s_{\mathscr{A}}]\!]$ that*

$$\left\{ x \in \mathcal{R}_s : \begin{pmatrix} x \\ x \end{pmatrix} \in V_s \right\} = \{0\}.$$

Proof. We just have to prove this result for $V_0 = V_{\boldsymbol{\Phi}}$. This result does not depend on the basis chosen for $\mathscr{C}$. We choose in this case the canonical basis where the matrix $\boldsymbol{\Phi}$ takes the following form

$$\begin{aligned} \boldsymbol{B}_u &\stackrel{\text{def}}{=} \begin{pmatrix} X^{(0)}_0 & X^{(0)}_1 & \cdots & X^{(0)}_{r-1-q^u} \\ X^{(0)}_{q^u} & X^{(0)}_{q^u+1} & \cdots & X^{(0)}_{r-1} \end{pmatrix} \\ \boldsymbol{\Phi} &\stackrel{\text{def}}{=} \left(\boldsymbol{B}^{(e)}_0 \middle| \boldsymbol{B}^{(e-1)}_1 \middle| \cdots \middle| \boldsymbol{B}^{(0)}_e \right) \\ \text{where } e &\stackrel{\text{def}}{=} e_{\mathscr{A}} \quad \text{(see (10))} \end{aligned}$$

Obviously since the variables involved in each block $\boldsymbol{B}^{(e-u)}_u$ are distinct it is sufficient to prove that for all u in $[\![0, e]\!]$ we have

$$\left\{ x \in \mathcal{R}_s : \begin{pmatrix} x \\ x \end{pmatrix} \in V_{\boldsymbol{\Phi}}(u) \right\} = \{0\},$$

where $V_{\mathbf{\Phi}}(u)$ is the column space of $\boldsymbol{B}_u$. To simplify notation let us drop the superscript (0) in $X_a^{(0)}$. Assume that there exists x in $\mathcal{R}$ and $\lambda_0, \cdots, \lambda_{r-1-q^u}$ in $\mathbb{F}_{q^m}$ such that

$$\sum_{i=0}^{r-1-q^u} \lambda_i \begin{pmatrix} X_i \\ X_{i+q^u} \end{pmatrix} = \begin{pmatrix} x \\ x \end{pmatrix} \tag{39}$$

This implies

$$\sum_{i=0}^{r-1-q^u} \lambda_i X_i = \sum_{i=0}^{r-1-q^u} \lambda_i X_{i+q^u},$$

which in turn implies that

$$\lambda_i = 0 \quad \text{where } i \in [\![0, q^u - 1]\!] \cup [\![r - q^u, r - 1]\!] \tag{40}$$

$$\lambda_i = \lambda_{i+q^u} \quad \text{where } i \in [\![0, r - 1 - 2q^u]\!]. \tag{41}$$

(40) and the shift invariance property (41) clearly imply that $\lambda_i = 0$ when $u \in [\![q^u, 2q^u - 1]\!]$. By induction we obtain that we should have $\lambda_i = 0$ for any $i \in [\![0, r - 1]\!]$. □

B A Heuristic for Computing less Pfaffians

We give here a heuristic for reducing the number of Pfaffians that are needed. When we compute a basis of the matrix code $\mathscr{C}_{\text{mat}}$, we obtain matrices of the following shape

$$\begin{pmatrix} & 1 & \\ 1 & & \\ & & \boxed{\boldsymbol{C}} \end{pmatrix}, \tag{42}$$

where $\boldsymbol{C}$ is a dense $i_0 \times i_0$ block whose size i_0 is the same for all basis elements and which is always located in the last i_0 row and column positions. There are only two 1's outside the block $\boldsymbol{C}$ of size $i_0 \times i_0$ and the rest of the values are equal to 0. This shape naturally comes from the fact that we see the space $(\boldsymbol{c}_{ij})_{i \le j} \in \mathbb{F}_q^{k(k+1)/2}$ such that $\sum_{i \le j} c_{ij} X_i X_j \in I_2(\mathscr{C}_s)$ as a linear code of length $k(k+1)/2$. The shape (42) then comes from the fact that we compute a basis of this code in systematic form, and then compute the matrices of the polar forms associated to the elements of this basis. This shape is also visible in the generic matrix $\boldsymbol{M}$ of the Pfaffian modeling: the coefficients of $\boldsymbol{M}$ that are outside this block area consist of single variables, while the coefficients in the dense area are dense linear forms.

Heuristic 2. *The Pfaffians obtained by taking a subset $\{i, j, k, l\}$ of rows and columns inside the dense area are linear combinations of the Pfaffians obtained by taking i outside the dense area.*

If the dense area is of size i_0, then Heuristic 2 implies that we only need to compute

$$\sum_{i=1}^{i_0-1} \binom{rm-s-i}{3}$$

Pfaffians, instead of $\binom{rm-s}{4}$. This is what we did to solve TII challenges 83, 89, 125, 166 and 210.

C The Shape of Rank 2 Matrices

We are going to prove Proposition 5.3 here and also give some additional explanation about the shape of rank 2 matrices in the matrix code of relationships. We will need again to introduce the canonical basis $\mathcal{A}$ of $\mathscr{C}$,

$$\mathcal{A} = (\boldsymbol{y}, \boldsymbol{x}\boldsymbol{y}, \ldots, \boldsymbol{x}^{r-1}\boldsymbol{y}, \ldots, \boldsymbol{y}^{2^{m-1}}, (\boldsymbol{x}\boldsymbol{y})^{2^{m-1}}, \ldots, (\boldsymbol{x}^{r-1}\boldsymbol{y})^{2^{m-1}}),$$

where $\boldsymbol{y} = \Gamma(\boldsymbol{x})^{-1}$. We define the matrix $\boldsymbol{H}_{\mathcal{A}}$ whose rows are the elements of $\mathcal{A}$ in that order. Let $\mathcal{B} = (\boldsymbol{b}_1, \ldots, \boldsymbol{b}_{rm})$ be the public basis of $\mathscr{C}$. As both $\boldsymbol{H}_{\mathcal{A}}$ and $\boldsymbol{H}_{\mathcal{B}}$ generate the same code $\mathscr{C}$, there exists a transition matrix $\boldsymbol{P} \in \mathbf{GL}_{rm}(\mathbb{F}_{2^m})$ such that $\boldsymbol{H}_{\mathcal{B}} = \boldsymbol{P}\boldsymbol{H}_{\mathcal{A}}$. Finally, we define

$$\boldsymbol{S} \stackrel{\text{def}}{=} \begin{pmatrix} & \boldsymbol{I}_r & \\ & \ddots & \\ \boldsymbol{I}_r & & \end{pmatrix}$$

the matrix of the r-th cyclic shift. We recall the following result.

Proposition C.1 ([CMT23b], Proposition 22). *The matrix codes $\mathscr{C}_{\mathrm{mat}}(\mathcal{A})$ and $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ are invariant under the linear operation*

$$\sigma : \boldsymbol{M} \longmapsto \boldsymbol{S}^{\top}\boldsymbol{M}^{(2)}\boldsymbol{S}.$$

Note that σ is rank-preserving. Therefore, the varieties given by matrices of rank at most 2 in $\mathscr{C}_{\mathrm{mat}}(\mathcal{A})$ and $\mathscr{C}_{\mathrm{mat}}(\mathcal{B})$ are also globally invariant under σ. This explains why Algorithm 2 returns m matrices of rank 2 instead of only one: it returns a whole σ-orbit $(\boldsymbol{B}, \sigma(\boldsymbol{B}), \ldots, \sigma^{m-1}(\boldsymbol{B}))$. Note that σ^m is the identity map.

The canonical basis is very useful, for rank 2 matrices with respect to this basis have a very special shape. We give the following as an assumption that has been verified through extensive computations.

Assumption 3. *Let $\boldsymbol{A} \in \mathscr{C}_{\mathrm{mat}}(\mathcal{A})$ be a matrix of rank 2. Up to changing $\boldsymbol{A}$ with $\sigma^j(\boldsymbol{A})$ for some j, we have that $\boldsymbol{A}$ is of the form*

$$\boldsymbol{A} = \left(\begin{array}{c|c} \boldsymbol{A}_0 & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{0} \end{array}\right), \tag{43}$$

with $\boldsymbol{A}_0$ a skew-symmetric matrix of size $e_{\mathscr{G}}r \times e_{\mathscr{G}}r$.

Remark C.2. When $\mathscr{C}$ is a dual alternant code whose support and multiplier are chosen independently at random, we experimentally observe that something similar can be said, with the matrix $\boldsymbol{A}_0$ having size $e_{\mathscr{A}} r \times e_{\mathscr{A}} r$ instead of $e_{\mathscr{G}} r \times e_{\mathscr{G}} r$.

This enables us to prove Proposition 5.3. Let us first recall it here.

Proposition 5.3. *Under Assumption 3, we have for some $a \in [\![0, m-1]\!]$:*

$$\boldsymbol{u}, \boldsymbol{v} \in \bigoplus_{j=a}^{a+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right).$$

Proof. We have $\boldsymbol{u} = \boldsymbol{\mu}\boldsymbol{H}_{\mathcal{B}} = \boldsymbol{\mu}\boldsymbol{P}\boldsymbol{H}_{\mathcal{A}}$ and $\boldsymbol{v} = \boldsymbol{\nu}\boldsymbol{H}_{\mathcal{B}} = \boldsymbol{\nu}\boldsymbol{P}\boldsymbol{H}_{\mathcal{A}}$. Define $\boldsymbol{A} = \boldsymbol{P}^\top \boldsymbol{B}\boldsymbol{P} \in \mathscr{C}_{\mathrm{mat}}(\mathcal{A})$. Since $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are chosen in the row span of $\boldsymbol{B}$, necessarily $\boldsymbol{\mu}\boldsymbol{P}$ and $\boldsymbol{\nu}\boldsymbol{P}$ are in the row span of $\boldsymbol{A}$. Indeed, if $\boldsymbol{\alpha} \in \mathbb{F}_{2^m}^{rm}$ is a vector such that $\boldsymbol{\alpha}\boldsymbol{B} = \boldsymbol{\mu}$, then

$$\boldsymbol{\alpha}(\boldsymbol{P}^\intercal)^{-1}\boldsymbol{A} = \boldsymbol{\alpha}\boldsymbol{B}\boldsymbol{P} = \boldsymbol{\mu}\boldsymbol{P},$$

which proves that $\boldsymbol{\mu}\boldsymbol{P}$ is in the row span of $\boldsymbol{A}$. The same reasoning holds for $\boldsymbol{\nu}\boldsymbol{P}$. Now, under Assumption 3, this implies that $\boldsymbol{\mu}\boldsymbol{P}$ and $\boldsymbol{\nu}\boldsymbol{P}$ belong to the image of the diagonal block $\boldsymbol{A}_0$, which we assume to be the upper-left block without loss of generality. As a result, one has $\boldsymbol{\mu}\boldsymbol{P}, \boldsymbol{\mu}\boldsymbol{P} \in \mathbb{F}_{2^m}^{e_{\mathscr{G}} r} \times \{0\}^{(m-e_{\mathscr{G}})r}$, from which we immediately deduce that $\boldsymbol{u}$ and $\boldsymbol{v}$ belong to the desired code. □

D Justification for the Algorithm Finding the Alignment between Triples

Heuristic 1 has enabled us to successfully reconstruct the desired code $\bigoplus_{j=a}^{a+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$ for all our implementations. To better understand its behaviour, we report on Table 6 the values that we observed for f when $r = 5$ and $m = 8$. The exact same values were observed for $3 \leq r \leq 7$.

i	0	1	2	3	4	5	6	7
$f(i)$	19	19	19	17	**15**	17	19	19
Aligned	No	No	No	No	**Yes**	No	No	No

Table 6: An example of the function f when $r = 5$ and $m = 8$.

The maximum of f is 19, which is $\binom{6+1}{2} - 2$. This means that the quadratic hull of $\mathscr{V}_i = \left\langle \boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}, \boldsymbol{u}'^{2^i}, \boldsymbol{v}'^{2^i}, \boldsymbol{w}'^{2^i} \right\rangle_{\mathbb{F}_q}$ contains only two relations, namely

$$\boldsymbol{u} \star \boldsymbol{v} = \boldsymbol{w}^2,$$

and

$$\boldsymbol{u}'^{2^i} \star \boldsymbol{v}'^{2^i} = \boldsymbol{w}'^{2^i}.$$

This is what we expect when i is equal to $b_0 + \lfloor m/2 \rfloor$, where b_0 is defined as the unique value (mod m) such that the triples are aligned. On the contrary, when $i = b_0$, all six vectors belong to $\bigoplus_{j=a}^{a+e_{\mathscr{G}}-1} \mathbf{GRS}_r\left(\boldsymbol{x}^{2^j}, \boldsymbol{y}^{2^j}\right)$, whose quadratic hull is much larger (this is due to the shape of quadratic relations in $\mathscr{C}_{\mathrm{rel}}(\mathcal{A})$ described in [FGO+11]). This implies that the six vectors are much likelier to satisfy additional quadratic relations, thus decreasing the value of f. This is how we find the value b_0 such that the two triples $(\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w})$ and $(\boldsymbol{u}'^{2^{b_0}}, \boldsymbol{v}'^{2^{b_0}}, \boldsymbol{w}'^{2^{b_0}})$ are aligned. Note that this reasoning also holds when $r = 2$, where it is worthwhile noticing that the minimum of f becomes 7. This new value for the minimum of f has a simple explanation: the corresponding code $\mathscr{V}_{b_0}$, provably, turns out to be a GRS code of dimension 4.

E The issue when $r = 5$

Our algorithm for $r = 5$ always found rank-2 matrices satisfying Assumption 3, with an additional annoying property: some rows and columns were always equal to 0. More precisely, we found that the diagonal block $\boldsymbol{A}_0$ of a rank-2 matrix $\boldsymbol{A}$ in the canonical basis had the following shape

$$\boldsymbol{A}_0 = \begin{pmatrix}
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&*&*&0&*&0&*&*&*&*&*&*&*&*&*&*&*&*\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&*&0&0&0&*&*&0&*&0&*&*&*&*&*&*&*&*&*&*&*\\
0&*&0&0&*&0&*&*&*&*&*&*&*&*&*&*&*&*&*&*&*\\
0&0&0&0&*&*&0&*&0&*&*&*&*&*&*&*&*&*&*&*&*\\
0&*&0&0&0&*&*&0&*&0&*&*&*&*&*&*&*&*&*&*&*\\
0&0&0&0&*&*&0&*&0&*&*&*&*&*&*&*&*&*&*&*&*\\
0&*&0&0&0&*&*&0&*&0&*&*&*&*&*&*&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&0&*&*&*&*&*&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&0&*&*&*&*&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&0&*&*&*&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&0&*&*&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&0&*&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&*&0&*&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&*&*&0&*&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&*&*&*&0&*&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&*&*&*&*&0&*&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&*&*&*&*&*&0&*\\
0&*&0&0&*&*&*&*&*&*&*&*&*&*&*&*&*&*&*&*&0
\end{pmatrix},$$

where the character $*$ denotes any element of $\mathbb{F}_{2^m}$. This block $\boldsymbol{A}_0$ not being entirely dense implied the vectors $\boldsymbol{u}_i, \boldsymbol{v}_i, \boldsymbol{w}_i$ belonging to a proper subcode of $\mathscr{D}$. We could still define the $\mathscr{D}'_i$s, but this led to $\mathscr{D}_{e_{\mathscr{G}}-1}$ being a 2-codimensional

subcode of $\mathbf{GRS}_r\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right)$. We have no explanation for the origin of this issue yet. However, we did find a way to recover a GRS code with support $\boldsymbol{x}$. The idea is to consider $\mathscr{D}_{e_{\mathscr{G}}-2}$, which is a subcode of

$$\mathbf{GRS}_r\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-2}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-2}}\right) \oplus \mathbf{GRS}_r\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right) = \mathbf{GRS}_{2r}\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right),$$

the equality coming from [Ran25, Lemma 12]. We always found that the codimension of $\mathscr{D}_{e_{\mathscr{G}}-2}$ in this GRS code was equal to 2. Our hope was that this codimension was small enough compared to $2r$ for $\mathscr{D}^{\star 2}_{e_{\mathscr{G}}-2}$ to generate all of

$$\mathbf{GRS}_{2r}\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}-1}}\right)^{\star 2} = \mathbf{GRS}_{4r-1}\left(\boldsymbol{x}^{2^{e_{\mathscr{G}}-1}}, \boldsymbol{y}^{2^{e_{\mathscr{G}}}}\right).$$

This was indeed the case in all our experiments. In particular, it enabled us to recover a valid pair of support and Goppa polynomial for the key recovery TII challenges 83 (which had already been solved by [Pan25]) and 89. Instances 125, 166 and 210 had parameters $r = 7, m = 9$ and did not require any trick: the algorithm described in the previous section worked well.