Groebner bases and error correcting codes: from Cooper Philosophy to Degrobnerization

Michela Ceria, Teo Mora, Massimiliano Sala

PCA 2020
(virtually in St. Petersburg)

Cooper's philosophy

Using Groebner bases computations in order to decode cyclic codes

Given some well determined set of polynomials, the lexicographical reduced Groebner basis is computed and employed for the decoding process, in order to detect and correct the errors eventually occurred during a transmission, by making some computations with the **locator polynomials**.

Ideas in this framework

Cooper has the idea to turn the problem to get the errors from the syndromes into a **problem on polynomials**. More precisely, Cooper takes a (finite) set of polynomials $\mathcal{F}_C$, such that the error locations are in $V(\mathcal{F}_C)$ and he computes the lexicographical reduced Groebner basis of $I = (\mathcal{F}_C)$. The required error locator polynomial can be directly computed via the elimination property of lexicographical Groebner bases.

Ideas in this framework

Chen et al.

- gave an approach to decoding via Newton identities, which was improved by Augot-Bardet-Faugere;
- introduced the so called **syndrome variety** and the related **syndrome ideal** and proposed to deduce via a Groebner basis pre-computation a series of polynomials from which they deduce the plain error locator polynomial for each error and associated syndromes. This approach has been refined by Loustaunau-York and Caboara-Mora.

Sala-Orsini

Many elements in the syndrome variety are **spurious** so not corresponding to any error vector. Orsini and Sala improved the decoding process by eliminating the spurious solutions of the system and introduced the **general error locator polynomial** (GELP).

GELP

Sala-Orsini's GELP is a polynomial $\sigma(z, s)$ such that, if the error correction capability of the considered cyclic code is t and $\mu \leq t$ errors occurred, then, given the corresponding syndrome vector $\bar{s}$, the roots of $\sigma(z, \bar{s})$ are the μ error locations and zero with multiplicity $t - \mu$.

Every cyclic code admits a GELP.

We should degroebnerize

In Sala-Orsini's context

- the input is not simple;
- it is computationally intensive to compute the Groebner basis;
- the output is huge;
- most of its elements are irrelevant (we only need the locator polynomial);
- and the locator can be very **dense**

The importance of being sparse

Decoding

Evaluating the GELP in the syndromes and finding the roots

The bottleneck in the decoding procedure, using the GELP, is the evaluation in the syndrome vector, it is useful to find a **sparse** version of such a polynomial and our analysis started from this point.

Our analysis

The code

$C\,[n, k, d]$ binary cyclic code (starting from BCH and then generalizing).
$t = 2$ so up to 2 error corrected.

Reversing the point of view

We study **Sala-Orsini** syndrome variety $V(\mathcal{F}_{OS})$ with an approach *à la Moeller*: we do not compute the Groebner basis from the polynomial system but we get the locator **via interpolation** from the variety.

Example: $n = 2^m - 1$ and $S = \{1, 3\}$

In this case $n = 2^m - 1$ and the primary defining set is $S = \{1, 3\}$.
We exclude the spurious elements in the syndrome variety:

$$\mathbf{X} = \{(c + d, c^3 + d^3, c, d),\ c, d \in \mathbb{F}_{2^m},\ c \neq d\}.$$

Variables: x_1, x_2 syndromes; z_1, z_2 locations; $x_1 < x_2 < z_1 < z_2$,
$\rightarrow (x_1, x_2, z_1, z_2)$.

Degroebnerization

Why you should not even think...

Groebner bases are not efficient to be computed, therefore **Degroebnerization** is aimed to **limit only to the very necessary cases the use of Groebner bases**, finding **alternative solutions** every time it is possible.

We will see in what follows, two fundamental tools for Degroebnerizing the decoding procedure.

First tool: Cerlienco-Mureddu correspondence

1990

Cerlienco and Mureddu study how to compute the lexicographical Groebner escalier of the (zerodimensional radical) ideal of a finite set of simple points $\mathbf{X} = \{P_1, ..., P_N\} \subset \mathbf{k}^n$, **without using Groebner bases**. They prove a bijection between the points and the monomials in the escalier.

The algorithm is purely **combinatoric** and only uses **comparisons** among the coordinates of the points. It is *iterative on the points* and *recursive on the variables*. Complexity $O(n^2N^2)$.

Ceria-Mora's version with Bar Codes

Complexity $O(nN^2 log(N))$.

Second tool: Marinari-Mora's theorem

Consider a zerodimensional radical ideal $I \triangleleft \mathcal{P}$, fixing on $\mathcal{P}$ the lexicographical order $<$ induced by $x_1 < x_2 < \ldots < x_n$. Denote by $\mathrm{N}(I)$ the associated lexicographical Groebner escalier and by

$$\mathrm{G}(I) := \{t_1, \ldots, t_r\} \subseteq \mathcal{T},\ t_i = x_1^{d_{i,1}} \cdots x_n^{d_{i,n}}$$

the monomial basis for the lexicographical semigroup ideal $\mathrm{T}(I)$. Then, there exist polynomials

$$\gamma_{m\delta i} = x_m - g_{m\delta i}(x_1, \ldots, x_{m-1}),$$

for each $i \in \{1, \ldots, r\}$, $m \in \{1, \ldots, n\}$ and $\delta \in \{1, \ldots, d_{i,m}\}$ such that the products

$$f_i = \prod_m \prod_\delta \gamma_{m\delta i},\ i = 1, \ldots, r$$

form a minimal Groebner basis of I, with respect to $<$.

Groebner escalier

Lemma

Let $\mathbf{X} = \{P_1, ..., P_N\}$ be a finite set of simple points in $\mathbf{k}^n$ and let d be the number of distinct elements in $\mathbf{k}$ that appear as first coordinate of some point in $\mathbf{X}$. Let $I(\mathbf{X}) \lhd \mathbf{k}[x_1, ..., x_n]$ be the ideal of points of $\mathbf{X}$ and $\mathrm{N}(\mathbf{X})$ its lexicographical Groebner escalier, supposing $x_1 < x_2 < ... < x_n$. Then it holds $1, x_1, x_1^2, ..., x_1^{d-1} \in \mathrm{N}(\mathbf{X})$.

Groebner escalier

Notation: if $\tau \in \mathcal{T}$ and $\mathsf{H} \subset \mathcal{T}$, then $\tau\mathsf{H} := \{\tau\sigma,\ \sigma \in \mathsf{H}\}$.

Theorem

Let $\mathsf{H} = \{1, x_1, ..., x_1^{q-2}\}$, where $q = n + 1 = 2^m$. The lexicographical Groebner escalier ($x_1 < x_2 < z_1 < z_2$) of the ideal $I = I(\mathbf{X})$ described as the ideal associated to
$\mathbf{X} = \{(c + d, c^l + d^l, c, d),\ c, d \in \mathbb{F}_{2^m},\ c, \neq d\}$ has the form

$$\mathsf{N}(I) = \mathsf{N}' \cup z_1\mathsf{N}',$$

where

$$\mathsf{N}' = \mathsf{H} \cup x_2\mathsf{H} \cup ... \cup x_2^{\frac{q}{2}-1}\mathsf{H}.$$

Only the case of two errors

If we want to study the case in which *exactly two errors* occur, we should remove from the variety the points of the form

$$(c, c^l, c, 0),\ (c, c^l, 0, c),$$

so the escalier becomes

$$\mathrm{N}(I) = \mathrm{N}' \cup z_1 \mathrm{N}',$$

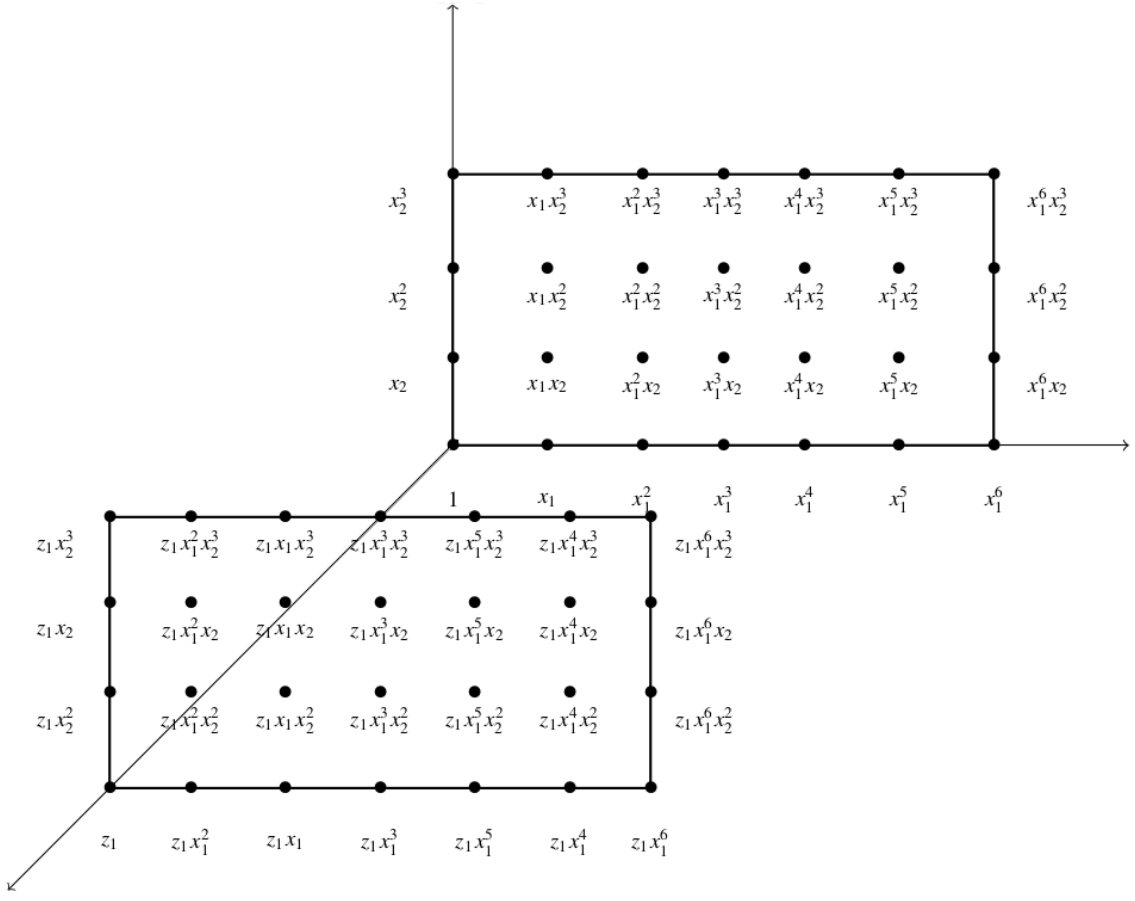
where $\mathrm{N}' = \mathrm{H} \cup x_2 \mathrm{H} \cup \ldots \cup x_2^{\frac{q}{2}-2} \mathrm{H}$.

Groebner escalier: visualizing it

If we identify each term $x_1^{\alpha_1} \cdots x_n^{\alpha_n} \in \mathcal{T}$ with its exponents' list $(\alpha_1, ..., \alpha_n) \in \mathbb{N}^n$ and we regards $(\alpha_1, ..., \alpha_n)$ as a point in the n-dimensional affine space, we can say that the escalier of ideal I has the shape of **two superimposed rectangles**.

Example: BCH over $\mathbb{F}_8$

$\mathbf{X} = \{(c + d, c^3 + d^3, c, d),\ c, d \in \mathbb{F}_8,\ c, \neq d\}$; the Groebner escalier $\mathrm{N}(I(\mathbf{X}))$ is given by $\mathrm{N}(I) = \mathrm{N}' \cup z_1\mathrm{N}'$, where
$\mathrm{N}' = \mathrm{H} \cup x_2\mathrm{H} \cup \ldots \cup x_2^3\mathrm{H}$ and $\mathrm{H} = \{1, x_1, \ldots, x_1^6\}$



The structure

- $z_1^2, z_2 \in \mathrm{G}(I)$ monomial basis
- $z_2 = z_1 + x_1 \in I \Rightarrow$ once known c, it is easy to deduce d (*linear* and *sparse*) $\Rightarrow$ only useful the polynomial with leading term z_1^2.

Marinari-Mora's theorem: $F_c := z_1 + f_c(x_1, x_2)$, $F_d := z_1 + f_d(x_1, x_2)$ and partition $\mathbf{X} = \mathbf{Z}_c \sqcup \mathbf{Z}_d$, $|\mathbf{Z}_c| = |\mathbf{Z}_d| = \frac{1}{2}|\mathbf{X}| = \binom{q}{2}$ s.t

- F_c zero on $\mathbf{Z}_c$; F_d zero on $\mathbf{Z}_d$;
- $(x_1, x_2, z_1, z_2) \in \mathbf{Z}_c \Leftrightarrow (x_1, x_2, z_2, z_1) \in \mathbf{Z}_d$.

The structure [2]

$\Rightarrow$ restriction to $\mathbf{Z}_c$, F_c: other locations from $z_2 = z_1 + x_1$. Therefore we interpolate in *half of the points*.

Half Error Locator Polynomial (HELP) is the polynomial F_c and it is the only polynomial really needed for decoding.

Decoding procedure

Preprocessing

Find a sparse HELP.

Step 1

Evaluate the HELP in the syndromes, finding a linear polynomial η in z_1.

Step 2

Solve in z_1 the equation $\eta = 0$; the root is one of the two error locations c.

Step 3

Evaluate the polynomial $z_2 + z_1 + x_1$ in the first syndrome ($x_1 = s_1$) and the first location ($z_1 = c$).

Step 4

Solve the equation $z_2 + s_1 + c = 0$ in z_2, getting the second location $d := s_1 + c$.

A sparsity matter

Pairs:
$[(c+d, c^3+d^3, c, d), (c+d, c^3+d^3, d, c)]$,
$\rightarrow$
Pick a point for each pair: the choice influences the sparsity of the HELP F_a

Example: inspection over $\mathbb{F}_8$

BCH code over $\mathbb{F}_8$

A bad choice

$$z_1 + x_1^3 x_2^3 + x_1^2 x_2^3 + a^4 x_1 x_2^3 + a^2 x_2^3 + a x_1^6 x_2^2 + a^6 x_1^5 x_2^2 + a^5 x_1^4 x_2^2 + a^6 x_1^3 x_2^2 + a^4 x_1^2 x_2^2 + a^6 x_1 x_2^2 + a^6 x_2^2 + a x_1^5 x_2 + a^4 x_1^4 x_2 + a^4 x_1^3 x_2 + x_1^2 x_2 + a^4 x_1 x_2 + x_2 + a^3 x_1^6 + a^5 x_1^5 + a^3 x_1^4 + a^2 x_1^3 + a^3 x_1^2 + a^2 x_1 + a^3$$

A good choice

$$z_1 + a^6 x_1^2 x_2^2 + a^4 x_1^5 x_2 + a^3 x_1$$

The difference is **in only one point!**

We need a good HELP

The "good choice" is still **not optimal**: it correct up to two errors but for the case of one error c, it returns the value 0, so to compute c we need the second equation $z_2 = z_1 + x_1$, namely we have $x_1 = c$, $z_1 = 0$ and we have to compute c as second location: $z_2 = 0 + c$. We will see soon that we can compute sparse HELPs giving directly the error c in this case.

Good choice - good pattern

A good choice

$z_1 + a^6 x_1^2 x_2^2 + a^4 x_1^5 x_2 + a^3 x_1$

x_1^6	0	0	0	0
x_1^5	0	a^4	0	0
x_1^4	0	0	0	0
x_1^3	0	0	0	0
x_1^2	0	0	a^6	0
x_1	a^3	0	0	0
1	0	0	0	0
	1	x_2	x_2^2	x_2^3

The knight gambit

A good choice

$z_1 + a^6 x_1^2 x_2^2 + a^4 x_1^5 x_2 + a^3 x_1$

$$\begin{pmatrix} x_1^6 & 0 & \mathbf{0} & 0 & 0 \\ x_1^5 & 0 & a^4 & \mathbf{0} & 0 \\ x_1^4 & 0 & 0 & \mathbf{0} & 0 \\ x_1^3 & 0 & 0 & \mathbf{0} & 0 \\ x_1^2 & 0 & 0 & a^6 & 0 \\ x_1 & \mathbf{a^3} & \mathbf{0} & 0 & 0 \\ 1 & 0 & \mathbf{0} & 0 & 0 \\ & 1 & x_2 & x_2^2 & x_2^3 \end{pmatrix}$$

Knight move: $x_1^{-3} x_2$

General structure for the case $n = 2^m - 1$ and $S = \{1, l\}$

The HELPs have at most $\frac{n+1}{2} + 1$ terms: $n = 2^m - 1$ length of the code.

General shape

$$\eta(x_1, x_2, z_1) = z_1 + \sum_{i=1}^{\frac{n+1}{2}} a_i x_1^{(n+1-li) \mod n} x_2^{(i-1) \mod \frac{n+1}{2}}$$

where $a_i \in GF(2^m)$ are the coefficients.

Knight move: $x_1^{-l} x_2$

What about the coefficients?

the HELP can be found performing Lagrange interpolation in the points with third coordinate $c = 1$ (with $\frac{d}{c} = a^{2i+1}$) plus the point $(1, 1, 1, 0)$ in the terms t^i, $0 \leq i \leq 2^{m-1}$, where $t = x_1^{-l \mod n} x_2$, that is, t is the knight move.

It has the form $\eta(x_1, x_2, z_1) = z_1 + x_1 g(t)$, where $g(t)$ is the Lagrange interpolator.

Example: BCH code over $\mathbb{F}_8$

For the code with $n = 7$ and $S = \{1, 3\}$ over $\mathbb{F}_8$, the HELP is $z_1 + x_1(x_1^5 x_2^3 + a^2 x_1 x_2^2 + a^4 x_1^4 x_2 + a)$.

Knight move:

$$
\begin{array}{cccccc}
x_1^6 & 0 & 0 & 0 & 1 \\
x_1^5 & 0 & a^4 & 0 & 0 \\
x_1^4 & 0 & 0 & 0 & 0 \\
x_1^3 & 0 & 0 & 0 & 0 \\
x_1^2 & 0 & 0 & a^2 & 0 \\
x_1 & a & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 \\
 & 1 & x_2 & x_2^2 & x_2^3
\end{array}
$$

It is easy to verify that the HELP corrects **up to two errors** and that if only one error occurs it is **directly returned** as output by the HELP.

HELP EXISTS AND IT CAN BE FOUND

Our aim is to decode a binary cyclic code C over $\mathbb{F}_{2^m}$, length $n = 2^m - 1$ and **primary** defining set $S_C = \{1, l\}$.
We have the $n(n-1)$ **non spurious points** (points composed by non spurious syndromes and the corresponding errors)

$$(c + d, c^l + d^l, c, d), c, d \in \mathbb{F}_{2^m}^*, c \neq d,$$

or, equivalently, $\binom{n}{2}$ pairs

$$\left\{ \Big(c + d, c^l + d^l, c, d\Big), \Big(c + d, c^l + d^l, d, c\Big) \right\}, c, d \in \mathbb{F}_{2^m}^*, c \neq d.$$

Moreover, we have to consider the n pairs of the form

$$\left\{ \Big(c, c^l, c, 0\Big), \Big(c, c^l, 0, c\Big) \right\}, c \in \mathbb{F}_{2^m}^*,$$

which correspond to the occurrence of one single error.

HELP exists and it can be found

Denoting by a any primitive element of $\mathbb{F}_{2^m}$ and setting $a^{-\infty} = 0$, we can represent these pairs as

$$\Big\{\Big(c(1+a^{2i+1}), c^l(1+a^{2li+l}), c, a^{2i+1}c\Big),$$
$$\Big(c(1+a^{2i+1}), c^l(1+a^{2li+l}), a^{2i+1}c, c\Big), c \in \mathbb{F}_{2^m}^*\Big\},$$
$$i \in \{0, \ldots, 2^{m-1}-1\} \cup \{-\infty\}.$$

setting $d/c := a^{2i+1}$ (in the case $d = 0$, $c \neq 0$, $a^{2i+1} = a^{-\infty}$).

HELP exists and it can be found

HELP, by construction, is the polynomial $\eta(x_1, x_2, z_1) = z_1 + h(x_1, x_2)$ such that if h is evaluated at each of the $\binom{n+1}{2}$ points

$$\left(c(1 + a^{2i+1}), c^l(1 + a^{2li+l})\right), c \in \mathbb{F}_{2^m}^*, i \in \{0, \ldots, 2^{m-1} - 1\} \cup \{-\infty\}$$

returns the value c.

HELP exists and it can be found

The Lagrange interpolator $g(t)$, $deg(g) = 2^{m-1} + 1$, which returns $(1 + a^{2i+1})^{-1}$ when evaluated at each values

$$t = (1 + a^{2i+1})^{-l}(1 + a^{2li+l}), i \in \{0, ..., 2^{m-1} - 1\} \cup \{-\infty\},$$

gives a HELP, in the sense that, defined $h(x_1, x_2) = x_1 g(x_1^{-l} x_2)$, it holds

$$\eta(x_1, x_2, z_1) = z_1 + h(x_1, x_2) = z_1 + x_1 g(x_1^{-l} x_2).$$

Proof

To prove that η is a HELP, we have to prove that, given a point

$$P = \left(c(1+a^{2i+1}), c^l(1+a^{2li+l})\right), c \in \mathbb{F}_{2^m}^*, i \in \{0, \ldots, 2^{m-1}-1\} \cup \{-\infty\}$$

it holds $h(P) = c$.

Proof

Note that $x_1 = c(1 + a^{2i+1})$ implies $c = x_1(1 + a^{2i+1})^{-1}$ and

$$x_2 = c^l(1 + a^{2li+l}) = x_1^l(1 + a^{2i+1})^{-l}(1 + a^{2li+l}).$$

Now, consider a point of the form
$P = \left(c(1 + a^{2i+1}), c^l(1 + a^{2li+l}), c, a^{2i+1}c\right)$,
$c \in \mathbb{F}_{2^m}^*$, $i \in \{0, \ldots, 2^{m-1} - 1\} \cup \{-\infty\}$ and evaluate $h(P)$:

$$h(P) = h(x_1, x_2) = x_1 g(x_1^{-l} x_2) = x_1 g(x_1^{-l} x_1^l (1 + a^{2i+1})^{-l}(1 + a^{2li+l}))$$
$$= c(1 + a^{2i+1})\underline{(\mathbf{1} + \mathbf{a^{2i+1}})^{-1}} = c.$$

This proves that η is a HELP.

No errors?

Our HELP is consistent also with the case in which no errors occur, even if **we do not consider the point $(0,0,0,0)$ in our variety**. Indeed, the HELP has shape $\eta(x_1, x_2, z_1) = z_1 + x_1 g(x_1, x_2)$.

When no error occurs, we have $x_1 = 0$, leading to $\eta = z_1$, giving the only root $z_1 = 0$. Since then $z_2 = z_1 + x_1$, it holds $z_2 = 0 + 0 = 0$ and so we retrieve the two zero locations.

What's going on now

In the case $n \mid 2^m - 1$, $S = \{1, l\}$, the escalier has a more involved form, even though the symmetry is respected.

- **Study of the structure**: a: primitive $(2^m - 1)^{\text{th}}$ root of unity, $\alpha := \frac{2^m - 1}{n}$ and $b := a^\alpha$ a primitive n^{th} root of unity,
  $\mathcal{R}_n := \{e \in \mathbb{F}_{2^m} : e^n = 1\}$ and $\mathcal{S}_n := \mathcal{R}_n \sqcup \{0\}$

  $$\begin{aligned}
  &Z_2 := \{(c + d, c^l + d^l, c, d), c, d \in \mathcal{R}_n, c \neq d\}, \#Z_2^\times = n^2 - n;\\
  &Z_+ := \{(c + d, c^l + d^l, c, d), c, d \in \mathcal{S}_n, c \neq d\}, \#Z_+^\times = n^2 + n,\\
  &Z_{ns} := \{(c + d, c^l + d^l, c, d), c, d \in \mathcal{S}_n\} \setminus \{(0, 0, c, c), c \in \\
  &\mathcal{R}_n\}, \#Z_{ns}^\times = n^2 + n + 1,\\
  &Z_e := \{(c + d, c^l + d^l, c, d), c, d \in \mathcal{S}_n\}, \#Z_e^\times = (n + 1)^2.
  \end{aligned}$$

- **The HELP in that case**, by means of Cerlienco-Mureddu correspondence.

The case $n \mid 2^m - 1$, $S = \{1, l\}$

Consider a binary cyclic code C with length $n \mid 2^m - 1$ and primary defining set $S = \{1, l\}$.
The syndrome variety of this code contains $n(n-1)$ *non spurious points* (namely points composed by non spurious syndromes and the corresponding errors)

$$(c + d, c^l + d^l, c, d), c, d \in \mathcal{R}_n, c \neq d,$$

or, equivalently, $\binom{n}{2}$ pairs

$$\left\{ \left(c + d, c^l + d^l, c, d \right), \left(c + d, c^l + d^l, d, c \right) \right\}, c, d \in \mathcal{R}_n, c \neq d, \quad (1)$$

where $\mathcal{R}_n$ denotes the set of n-th roots of unity in $\mathbb{F}_{2^m}$. Moreover, we have to consider the n pairs of the form

$$\left\{ \left(c, c^l, c, 0 \right), \left(c, c^l, 0, c \right) \right\}, c \in \mathcal{R}_n, \quad (2)$$

which correspond to the occurrence of one single error. In total, we have $\binom{n+1}{2}$ pairs, corresponding to the occurrence of one or two errors.

Representing the points

Denoting by a any primitive element of $\mathbb{F}_{2^m}$ and setting $a^{-\infty} = 0$, we can represent these pairs as

$$\left\{ \left(c(1 + a^{2i+1}), c^l(1 + a^{2li+l}), c, a^{2i+1}c \right), \right.$$
$$\left. \left(c(1 + a^{2i+1}), c^l(1 + a^{2li+l}), a^{2i+1}c, c \right), c \in \mathcal{R}_n \right\},$$
$$i \in \{0, \ldots, 2^{m-1} - 1\} \cup \{-\infty\}.$$

setting $d/c := a^{2i+1}$ (in the case $d = 0$, $c \neq 0$, $a^{2i+1} = a^{-\infty}$).

Who is the HELP?

HELP, by construction, is the polynomial $\eta(x_1, x_2, z_1) = z_1 + h(x_1, x_2)$ such that if h is evaluted on each of the $\binom{n+1}{2}$ points

$$\left(c(1 + a^{2i+1}), c^l(1 + a^{2li+l})\right), c \in \mathcal{R}_n, i \in \{0, ..., 2^{m-1} - 1\} \cup \{-\infty\} \tag{3}$$

returns the value c.

The escalier is not so simple

In order to get the HELP we need, we consider the set $\mathcal{Q}$, given by the $\frac{n+1}{2}$ points

$$\mathcal{Q} := \left\{ \left((1 + a^{2i+1}), (1 + a^{2li+l}) \right),\ i \in \{0, \ldots, 2^{m-1} - 1\} \cup \{-\infty\} \right\},$$

so we take the points of the syndrome variety with $c = 1$ and we consider only the first and the second coordinates, namely the syndromes, keeping in mind that **a correctable syndrome vector uniquely identifies an error vector.**

The escalier is not so simple

We make now the following coordinates' change:

$$\begin{cases} X = x_1^n \\ Y = x_1^{-l} x_2, \end{cases}$$

under which, the points in $\mathcal{Q}$ become of the form:

$$\mathcal{Q}' := \Big\{ \Big((1+a^{2i+1})^n, (1+a^{2i+1})^{-l}(1+a^{2li+l}) \Big), \\ i \in \{0, \ldots, 2^{m-1}-1\} \cup \{-\infty\} \Big\}.$$

The escalier is not so simple

The set $\mathcal{Q}'$ is finite, in particular $|\mathcal{Q}'| \leq |\mathcal{Q}| = \frac{n+1}{2}$, and we can perform Cerlienco-Mureddu correspondence on it, getting $\mathsf{N}(I(\mathcal{Q}')) := \Phi(\mathcal{Q}')$, the lexicographical Groebner escalier of the ideal associated to the variety $\mathcal{Q}'$. This order ideal contains $|\mathcal{Q}'|$ terms.

As proved by Cerlienco-Mureddu, there exists a polynomial $g'(X, Y) = \sum_{t \in \mathsf{N}(I(\mathcal{Q}'))} b_t t$, $b_t \in \mathbb{F}_{2^m}$, that **takes value** $(1 + a^{2i+1})^{-1}$ **at each point in** $\mathcal{Q}'$.

Back to $\mathcal{Q}$

Changing back the coordinates from (X, Y) to (x_1, x_2), we get a polynomial $g(x_1, x_2)$ taking value $(1 + a^{2i+1})^{-1}$ once evaluated in the elements of $\mathcal{Q}$.

HELP exists, even with $n \mid 2^m - 1$.

The polynomial

$$\eta(x_1, x_2, z_1) := z_1 + h(x_1, x_2),$$

where $h(x_1, x_2) = x_1 g(x_1, x_2)$ is a HELP.

Proof: aim

To prove that η is a HELP, we have to prove that, given a point

$$\left(c(1+a^{2i+1}), c^l(1+a^{2li+l})\right), c \in \mathcal{R}_n,$$
$$i \in \{0, \ldots, 2^{m-1}-1\} \cup \{-\infty\}$$

it holds $h(P) = c$.

Proof: shape of the HELP

By construction, the polynomial $h(x_1, x_2)$ has the form:

$$h(x_1, x_2) = \sum_{j=0}^{r-1} x_1^{nj+1} \sum_{k=0}^{s_j-1} b_{jk} (x_1^{-l} x_2)^k,$$

where r is the number of different first coordinates of the points in $\mathcal{Q}'$ and s_j is the number of occurrences in $\mathcal{Q}'$ of the $(j+1)$-th first coordinate $j = 0, \ldots, r-1$.

Proof: evaluation [1]

Now, to conclude, only an evaluation is needed:

$$\sum_{j=0}^{r-1} c^{nj+1}(1+a^{2i+1})^{nj+1} \sum_{k=0}^{s_j-1} b_{jk} c^{-lk}(1+a^{2i+1})^{-lk} c^{lk}(1+a^{2li+l}) =$$

$$c \sum_{j=0}^{r-1} c^{nj}(1+a^{2i+1})^{nj+1} \sum_{k=0}^{s_j-1} b_{jk} c^{-lk}(1+a^{2i+1})^{-lk} c^{lk}(1+a^{2li+l})$$

Proof: evaluation [2]

Since $c \in \mathcal{R}_n$, $c^n = 1$, so

$$c \sum_{j=0}^{r-1} c^{nj}(1+a^{2i+1})^{nj+1} \sum_{k=0}^{s_j-1} b_{jk} c^{-lk}(1+a^{2i+1})^{-lk} c^{lk}(1+a^{2li+l}) =$$

$$c(1+a^{2i+1}) \sum_{j=0}^{r-1} (1+a^{2i+1})^{nj} \sum_{k=0}^{s_j-1} b_{jk}(1+a^{2i+1})^{-lk}(1+a^{2li+l}) =$$

$$c(x_1 g)((1+a^{2i+1}),(1+a^{2li+l})) = c,$$

where the last equality comes from the fact that $g(P) = (1+a^{2i+1})^{-1}$, for each $P' \in \mathcal{Q}$ and so $x_1 g$ takes value one once evaluated in P.

Thank you
for your attention!