Computational aspects of Algebraic Geometry codes

Grigory Solomatov (grigorys93@gmail.com)

Technical University of Denmark

October 2018-2021

Supervisors: Johan Rosenkilde, Peter Beelen

DTU Compute
Department of Applied Mathematics and Computer Science

Publications

- **Beelen, Rosenkilde, Solomatov**
  Fast encoding of AG codes over $C_{a,b}$-curves
  IEEE Transactions on Information Theory 2021
- **Neiger, Rosenkilde, Solomatov**
  Generic bivariate multi-point evaluation, interpolation and modular composition with precomputation
  International Symposium on Symbolic and Algebraic Computation 2020
- **Puchinger, Rosenkilde, Solomatov**
  Improved Power Decoding of Algebraic Geometry Codes
  IEEE International Symposium on Information Theory 2021
- **Beelen, Rosenkilde, Solomatov**
  Fast list decoding of Algebraic Geometry codes
  Submitted to IEEE Transactions on Information Theory 2022

Outline

- Why care?
- How to encode AG codes?
- How to decode AG codes?

Outline

- Why care?
- How to encode AG codes?
- How to decode AG codes?

Error-correcting codes

Algebraic Geometry codes

Alphabet = finite field, here $\mathbb{F} = \mathbb{F}_{29} = \mathbb{Z}/\langle 29 \rangle$

Simple case: Reed-Solomon codes (1960s)

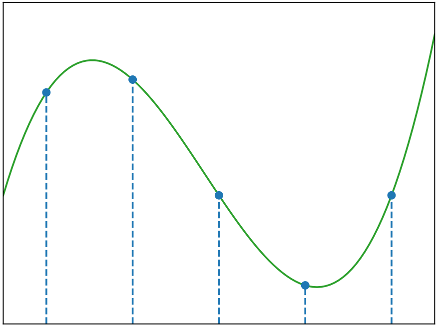
- message:

$$\begin{array}{ccccccccccccc} & W & & I & & Z & & A & & R & & D \\ & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ & 23 & & 9 & & 26 & & 1 & & 18 & & 4 \\ & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ f(x) = & 23 & + & 9x & + & 26x^2 & + & 1x^3 & + & 18x^4 & + & 4x^5 \end{array}$$

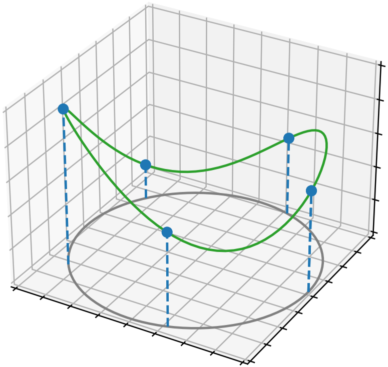
- codeword:

$$\boldsymbol{c} = \Big(f(0)\ ,\ f(1)\ ,\ f(2)\ ,\ \ldots\ ,\ f(28) \Big) \in \mathbb{F}^{29}$$

Algebraic Geometry codes



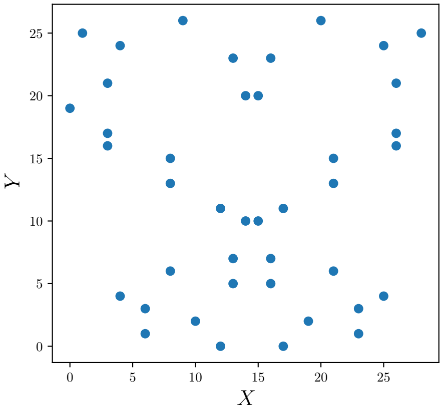
RS codes



AG codes

Algebraic curves?

$$x^{26} + y^{25} + 2y^2 + 1 = 0 \text{ (41 points)}$$



Computational aspects?

Encoding: given f, compute $\boldsymbol{c} = \Big(\; f(P_1) \; , \; f(P_2) \; , \; f(P_3) \; , \; \ldots \; , \; f(P_n) \; \Big) \in \mathbb{F}^n$

Decoding: given $\boldsymbol{r} = (\; r_1 \; , \; r_2 \; , \; r_3 \; , \; \ldots \; , \; r_n \;)$, recover f

Computational aspects?

Encoding: given f, compute $\boldsymbol{c} = \Big(f(P_1)\ ,\ f(P_2)\ ,\ f(P_3)\ ,\ \ldots\ ,\ f(P_n) \Big) \in \mathbb{F}^n$

Decoding: given $\boldsymbol{r} = (\ r_1\ ,\ r_2\ ,\ r_3\ ,\ \ldots\ ,\ r_n\ )$, recover f

Efficiency

Time, energy, memory, operations in $\mathbb{F}$ $(\ +\ ,\ -\ ,\ \cdot\ ,\ \div\ )$

Outline

- Why care?
- How to encode AG codes?
- How to decode AG codes?

Encoding over $C_{a,b}$-curves

- **Beelen, Rosenkilde, Solomatov**: *Fast encoding of AG codes over $C_{a,b}$-curves*
  IEEE Transactions on Information Theory 2021



$C_{a,b}$-curves

Definition

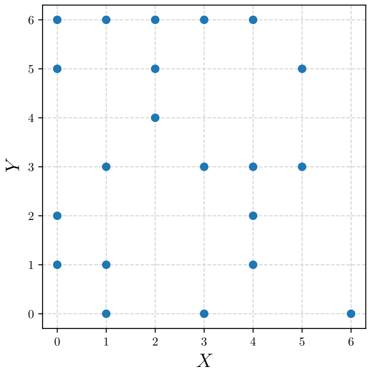
For $a, b \in \mathbb{Z}_{>0}$ coprime, a $C_{a,b}$-curve is defined by a polynomial $H \in \mathbb{F}[x, y]$ satisfying

1. $x^b, y^a \in \operatorname{supp} H$,
2. $x^i y^j \in \operatorname{supp} H \implies ai + bj \leq ab$,
3. no singularities $\langle H, \frac{\partial H}{\partial x}, \frac{\partial H}{\partial y} \rangle_{\mathbb{F}[x,y]} = \mathbb{F}[x, y]$

$C_{a,b}$-curves

$\mathbb{F} = \mathbb{F}_7$

Example



$$
\begin{aligned}
H(x,y) = \;& y^4 \\
& +4xy^3 \\
& +2y^2 + xy^2 + 3x^2y^2 \\
& +2xy + 3x^2y + 5x^3y \\
& + 4 + x + 6x^2 + 5x^3 + 4x^4 + x^5
\end{aligned}
$$

Encoding over $C_{a,b}$-curves

$\mathbb{F} = \mathbb{F}_{11}$

Problem

Given

- message: $$f(x,y) = \begin{matrix} 1x^0y^4 & + & 2x^1y^4 & + & 3x^2y^4 & + & 4x^3y^4 & + & 5x^4y^4 \\ 2x^0y^3 & + & 3x^1y^3 & + & 4x^2y^3 & + & 5x^3y^3 & + & 6x^4y^3 \\ 3x^0y^2 & + & 4x^1y^2 & + & 5x^2y^2 & + & 6x^3y^2 & + & 7x^4y^2 \\ 4x^0y^1 & + & 5x^1y^1 & + & 6x^2y^1 & + & 7x^3y^1 & + & 8x^4y^1 \\ 5x^0y^0 & + & 6x^1y^0 & + & 7x^2y^0 & + & 8x^3y^0 & + & 9x^4y^0 \end{matrix}$$
- points: $(x_1,y_1)\ ,\ (x_2,y_2)\ ,\ \ldots\ ,\ (x_n,y_n)$

compute the codeword: $\boldsymbol{c} = \Big(\ f(x_1,y_1)\ ,\ f(x_2,y_2)\ ,\ \cdots\ f(x_n,y_n)\ \Big)$.

Encoding over $C_{a,b}$-curves

$\mathbb{F} = \mathbb{F}_{11}$

Problem

Given

- message: $f(x,y) = \begin{array}{ccccccccc} 1x^0y^4 & + & 2x^1y^4 & + & 3x^2y^4 & + & 4x^3y^4 & + & 5x^4y^4 \\ 2x^0y^3 & + & 3x^1y^3 & + & 4x^2y^3 & + & 5x^3y^3 & + & 6x^4y^3 \\ 3x^0y^2 & + & 4x^1y^2 & + & 5x^2y^2 & + & 6x^3y^2 & + & 7x^4y^2 \\ 4x^0y^1 & + & 5x^1y^1 & + & 6x^2y^1 & + & 7x^3y^1 & + & 8x^4y^1 \\ 5x^0y^0 & + & 6x^1y^0 & + & 7x^2y^0 & + & 8x^3y^0 & + & 9x^4y^0 \end{array}$
- points: $(x_1, y_1)\ ,\ (x_2, y_2)\ ,\ \ldots\ ,\ (x_n, y_n)$

compute the codeword: $\boldsymbol{c} = \Big(\ f(x_1, y_1)\ ,\ f(x_2, y_2)\ ,\ \cdots\ f(x_n, y_n)\ \Big)$.

– Naive approach: $\mathcal{O}(n^2)$ operations in $\mathbb{F}$.

Encoding over $C_{a,b}$-curves

$$
\begin{array}{ccccccccc}
1x^0y^4 & + & 2x^1y^4 & + & 3x^2y^4 & + & 4x^3y^4 & + & 5x^4y^4 \\
2x^0y^3 & + & 3x^1y^3 & + & 4x^2y^3 & + & 5x^3y^3 & + & 6x^4y^3 \\
3x^0y^2 & + & 4x^1y^2 & + & 5x^2y^2 & + & 6x^3y^2 & + & 7x^4y^2 \\
4x^0y^1 & + & 5x^1y^1 & + & 6x^2y^1 & + & 7x^3y^1 & + & 8x^4y^1 \\
5x^0y^0 & + & 6x^1y^0 & + & 7x^2y^0 & + & 8x^3y^0 & + & 9x^4y^0
\end{array}
$$

Encoding over $C_{a,b}$-curves

$$
\begin{aligned}
&(1x^0 + 2x^1 + 3x^2 + 4x^3 + 5x^4)y^4 \\
&(2x^0 + 3x^1 + 4x^2 + 5x^3 + 6x^4)y^3 \\
&(3x^0 + 4x^1 + 5x^2 + 6x^3 + 7x^4)y^2 \\
&(4x^0 + 5x^1 + 6x^2 + 7x^3 + 8x^4)y^1 \\
&(5x^0 + 6x^1 + 7x^2 + 8x^3 + 9x^4)y^0
\end{aligned}
$$

Encoding over $C_{a,b}$-curves

$$
\begin{array}{ccccc}
0 & 1 & 2 & 3 & 4 \\
\shortparallel & \shortparallel & \shortparallel & \shortparallel & \shortparallel \\
x_1 & x_2 & x_3 & x_4 & x_5
\end{array}
$$

$$(\ 1\ ,\ 4\ ,\ 8\ ,\ 8\ ,\ 9\ )y^4$$

$$(\ 2\ ,\ 9\ ,\ 6\ ,\ 8\ ,\ 9\ )y^3$$

$$(\ 3\ ,\ 3\ ,\ 4\ ,\ 8\ ,\ 9\ )y^2$$

$$(\ 4\ ,\ 8\ ,\ 2\ ,\ 8\ ,\ 9\ )y^1$$

$$(\ 5\ ,\ 2\ ,\ 0\ ,\ 8\ ,\ 9\ )y^0$$

Encoding over $C_{a,b}$-curves

0	1	2	3	4
$\shortparallel$	$\shortparallel$	$\shortparallel$	$\shortparallel$	$\shortparallel$
x_1	x_2	x_3	x_4	x_5
$1y^4$	$4y^4$	$8y^4$	$8y^4$	$9y^4$
$+$	$+$	$+$	$+$	$+$
$2y^3$	$9y^3$	$6y^3$	$8y^3$	$9y^3$
$+$	$+$	$+$	$+$	$+$
$3y^2$	$3y^2$	$4y^2$	$8y^2$	$9y^2$
$+$	$+$	$+$	$+$	$+$
$4y^1$	$8y^1$	$2y^1$	$8y^1$	$9y^1$
$+$	$+$	$+$	$+$	$+$
$5y^0$	$2y^0$	$0y^0$	$8y^0$	$9y^0$

Encoding over $C_{a,b}$-curves

$0 = x_1$	$1 = x_2$	$2 = x_3$	$3 = x_4$	$4 = x_5$
5	2	0	8	9
4	4	9	7	1
2	1	9	6	4
3	4	5	0	0
2	10	7	0	0

Complexity

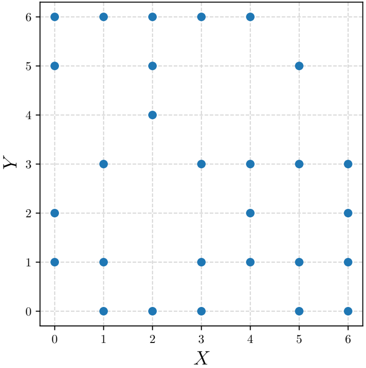
- $\widetilde{\mathcal{O}}(n)$ for semi-grids

Equal number of y-coordinates for each x-coordinate

Complexity

- $\widetilde{\mathcal{O}}(n)$ for semi-grids

Equal number of y-coordinates for each x-coordinate



Complexity

- $\widetilde{\mathcal{O}}(n)$ for semi-grids

Equal number of y-coordinates for each x-coordinate

- $\widetilde{\mathcal{O}}(n^{5/4})$ for curves close to the Hasse-Weil bound

$n \leq 2g\sqrt{q} + (q+1)$

Complexity

- $\widetilde{\mathcal{O}}(n)$ for semi-grids — Equal number of y-coordinates for each x-coordinate
- $\widetilde{\mathcal{O}}(n^{5/4})$ for curves close to the Hasse-Weil bound — $n \leq 2g\sqrt{q} + (q+1)$
- $\widetilde{\mathcal{O}}(n^{3/2})$ for reasonable curves — $n \geq q$

Complexity

- $\widetilde{\mathcal{O}}(n)$ for semi-grids — Equal number of y-coordinates for each x-coordinate
- $\widetilde{\mathcal{O}}(n^{5/4})$ for curves close to the Hasse-Weil bound — $n \leq 2g\sqrt{q} + (q+1)$
- $\widetilde{\mathcal{O}}(n^{3/2})$ for reasonable curves — $n \geq q$
- $\widetilde{\mathcal{O}}(an_x)$ in general — $a = \deg_y H, \quad n_x = \#x$-coordinates

Alternative approach: Reshaping

- **Neiger, Rosenkilde, Solomatov**
  Generic bivariate multi-point evaluation, interpolation and modular composition with precomputation
  International Symposium on Symbolic and Algebraic Computation 2020

Alternative approach: Reshaping

$\mathbb{F} = \mathbb{F}_{11}$

Problem

Given

- message: $f(x,y) = \begin{array}{ccccccccc} 1x^0y^4 & + & 2x^1y^4 & + & 3x^2y^4 & + & 4x^3y^4 & + & 5x^4y^4 \\ 2x^0y^3 & + & 3x^1y^3 & + & 4x^2y^3 & + & 5x^3y^3 & + & 6x^4y^3 \\ 3x^0y^2 & + & 4x^1y^2 & + & 5x^2y^2 & + & 6x^3y^2 & + & 7x^4y^2 \\ 4x^0y^1 & + & 5x^1y^1 & + & 6x^2y^1 & + & 7x^3y^1 & + & 8x^4y^1 \\ 5x^0y^0 & + & 6x^1y^0 & + & 7x^2y^0 & + & 8x^3y^0 & + & 9x^4y^0 \end{array}$

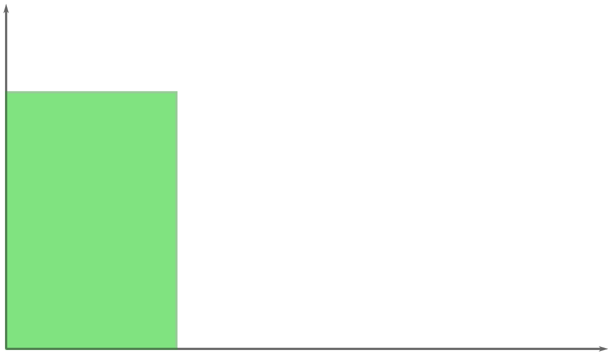
- points: $(x_1, y_1)\ ,\ (x_2, y_2)\ ,\ \ldots\ ,\ (x_n, y_n)$ with distinct x-coordinates

compute the codeword: $\boldsymbol{c} = \Big(\ f(x_1, y_1)\ ,\ f(x_2, y_2)\ ,\ \ldots\ ,\ f(x_n, y_n)\ \Big)$

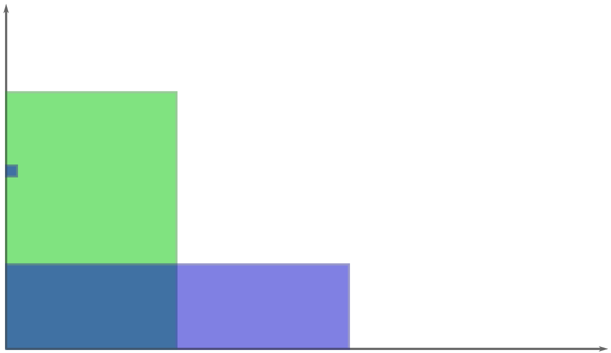
Strategy

$$f(x,y) \to h(x) \to \boldsymbol{c} = \Big(\ h(x_1)\ ,\ h(x_2)\ ,\ \ldots\ ,\ h(x_n) \ \Big)$$

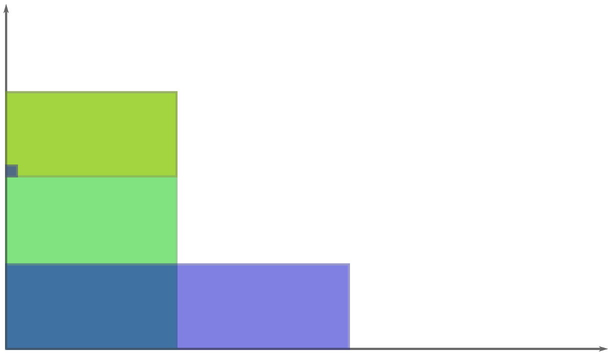
Reshaping



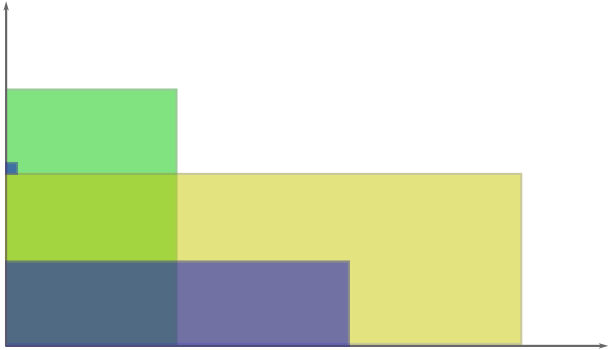
Reshaping



Reshaping



Reshaping



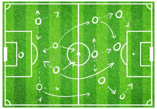
Reshaping

Reshaping

Cost: $\widetilde{\mathcal{O}}(n)$ if the points are

1. generic/random

2. available for precomputation



Outline

- Why care?
- How to encode AG codes?
- How to decode AG codes?

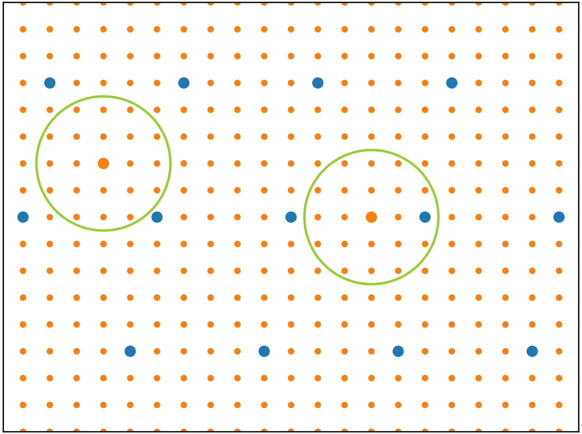
List decoding

- **Beelen, Rosenkilde, Solomatov**: *Fast list decoding of Algebraic Geometry codes*
  Submitted to IEEE Transactions on Information Theory



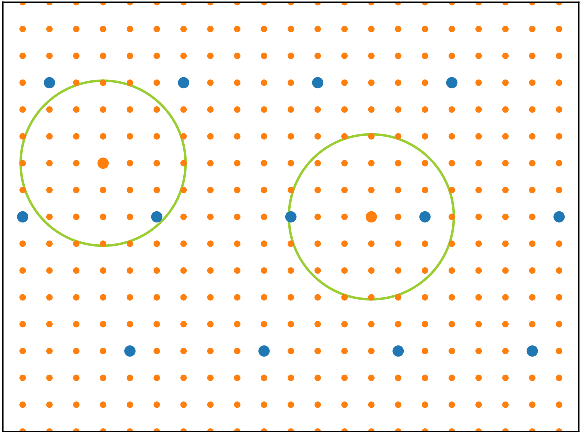
List decoding

Hamming distance: $d(\boldsymbol{w}_1, \boldsymbol{w}_2) =$ number of differing entries



List decoding

Hamming distance: $d(\boldsymbol{w}_1, \boldsymbol{w}_2) =$ number of differing entries



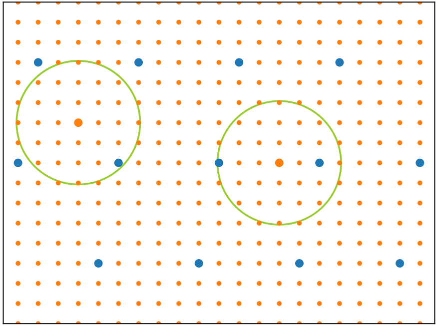
How to list decode?

Theorem (Guruswami-Sudan)

If we can find a $Q = Q_0 + Q_1 z + \cdots + Q_\ell z^\ell \in F[z]$ satisfying certain size and interpolation constraints then $Q(f) = 0$.

Decoding strategy

1. **Interpolation step:** compute Q
2. **Root-finding step:** find all roots of Q



How to list decode *fast*?

Interpolation step

1. Find a well-behaved function $x \in F$.
2. Express everything as $\mathbb{F}[x]$-modules.
3. Use existing fast algorithms for matrices over $\mathbb{F}[x]$.

Root-finding step

1. Convert $Q \in F[z]$ to $\mathbb{F}[[x]][z]$.
2. Use existing fast algorithms for root-finding over $\mathbb{F}[[x]]$.
3. Convert the found $\mathbb{F}[[x]]$-roots back to F.

Results

- Can decode *any* AG code!
- Faster than *any* other general decoder!
- At least as fast as *any* specialized decoder!

(except for RS codes)

Results

Cost: $\widetilde{\mathcal{O}}\Big(s\ell^{\omega}\mu^{\omega-1}(n+g)\Big) \subseteq \widetilde{\mathcal{O}}(\ell^4\mu^2 n)$

- $\ell =$ list size
- $s =$ multiplicity
- $\mu =$ smallest pole order at P_∞
- $\omega =$ matrix multiplication exponent
- $n =$ code length
- $g =$ genus

Previous work:

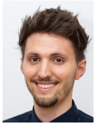
- $C_{a,b}$-curves: $\widetilde{\mathcal{O}}(\ell^5\mu^3(n+g))$ – **Beelen, Brander (2010)**
- General curves: $\widetilde{\mathcal{O}}(\mu n^2)$ – **Sakata, Fujisawa (2014)**
- One-point Hermitian: $\widetilde{\mathcal{O}}(s\ell^{\omega}n^{5/3})$ – **Rosenkilde, Beelen (2015)**
- Reed-Solomon: $\widetilde{\mathcal{O}}(s^2\ell^{\omega-1}n)$ – **Chowdhury, Jeannerod, Neiger, Schost (2015)**

Summary

- fast encoding over $C_{a,b}$-curves
- fast "encoding" over random points
- fast decoding of *all* AG codes
- power decoding works for (almost) all AG codes

Power-decoding

- **Puchinger, Rosenkilde, Solomatov**: *Improved Power Decoding of Algebraic Geometry Codes*
  IEEE International Symposium on Information Theory 2021

Power-decoding

- message: $f \in \mathcal{L}(G)$
- evaluation points: $D = P_1 + P_2 + \cdots + P_n$
- codeword: $\boldsymbol{c} = \big(f(P_1), f(P_2), \ldots, f(P_n)\big)$
- received word: $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$
- error-locator: $\Lambda_s(P_j) = 0$ for every error position P_j (with multiplicity s)
- interpolator: $R(P_1) = r_1$, $R(P_2) = r_2$, …, $R(P_n) = r_n$

$$\Lambda_s f^t - \sum_{j=0}^{\min\{t,s-1\}} \binom{t}{j} \Lambda_s (f - R)^j R^{t-j} \in \begin{cases} \{0\} & \text{if } 1 \leq t \leq s-1 \\ \mathcal{L}(\lambda_s P_\infty + t(G + \rho P_\infty) - sD) & \text{if } s \leq t \leq \ell \end{cases}$$

Power-decoding

- message: $f \in \mathcal{L}(G)$
- evaluation points: $D = P_1 + P_2 + \cdots + P_n$
- codeword: $\boldsymbol{c} = \big(f(P_1), f(P_2), \ldots, f(P_n)\big)$
- received word: $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$
- error-locator: $\Lambda_s(P_j) = 0$ for every error position P_j (with multiplicity s)
- interpolator: $R(P_1) = r_1$, $R(P_2) = r_2$, $\ldots$, $R(P_n) = r_n$

$$\underbrace{\Lambda_s f^t}_{\phi_t} - \sum_{j=0}^{\min\{t,s-1\}} \binom{t}{j} \underbrace{\Lambda_s (f-R)^j}_{\psi_j} R^{t-j} \in \begin{cases} \{0\} & \text{if } 1 \le t \le s-1 \\ \mathcal{L}\big(\lambda_s P_\infty + t(G + \rho P_\infty) - sD\big) & \text{if } s \le t \le \ell \end{cases}$$

Power-decoding

- message: $f \in \mathcal{L}(G)$
- evaluation points: $D = P_1 + P_2 + \cdots + P_n$
- codeword: $\boldsymbol{c} = \big(f(P_1), f(P_2), \ldots, f(P_n)\big)$
- received word: $\boldsymbol{r} = (r_1, r_2, \ldots, r_n)$
- error-locator: $\Lambda_s(P_j) = 0$ for every error position P_j (with multiplicity s)
- interpolator: $R(P_1) = r_1 \ , R(P_2) = r_2 \ , \ \ldots, \ R(P_n) = r_n$

$$\underbrace{\Lambda_s f^t}_{\phi_t} - \sum_{j=0}^{\min\{t,s-1\}} \binom{t}{j} \underbrace{\Lambda_s (f - R)^j}_{\psi_j} R^{t-j} \in \begin{cases} \{0\} & \text{if } 1 \leq t \leq s-1 \\ \mathcal{L}(\lambda_s P_\infty + t(G + \rho P_\infty) - sD) & \text{if } s \leq t \leq \ell \end{cases}$$

$\tau \to n(1 - \sqrt{\deg G / n})$ (if $2g - 1 \leq \deg G < n$)