

## 1 Codes de Reed-Solomon (vers 1960): appréciez l'élégance!

Soit  $k \in [1, n], \mathbb{F}_q$  tel que  $n \leq q$  et  $\alpha_1, \alpha_2, \dots, \alpha_n$  des "points d'évaluation" distincts de  $\mathbb{F}_q$ . A un message on associe un polynôme:

$$m = (m_0, m_1, \dots, m_{k-1}) \leftrightarrow f_m(X) = \sum_{i=0}^{k-1} m_i X^i.$$

Le code de Reed-Solomon (RS) est

$$C = \{RS(m) = (f_m(\alpha_1), f_m(\alpha_2), \dots, f_m(\alpha_n)) : f_m(X) \in \mathbb{F}_q[X], \deg(f) < k\}$$

On observe que pour tout message  $m$  et  $m'$

$$f_m(X) + f_{m'}(X) = f_{m+m'}(X)$$

et

$$a \cdot f_m(X) = f_{a \cdot m}(X)$$

et donc (comme  $\deg(f_{m+m'}(X)) < k$ )

$$RS(m) + RS(m') \in C$$

et

$$a \cdot RS(m) \in C.$$

Un code RS est donc linéaire. Alternativement, la linéarité se voit car l'encodage correspond à

$$(x_1, x_2, \dots, x_n) = (m_0, m_1, \dots, m_{k-1}) \begin{pmatrix} 1 & 1 & 1 & \dots & \dots & 1 \\ \alpha_1 & \alpha_2 & \dots & \dots & \dots & \alpha_n \\ \alpha_1^2 & \alpha_2^2 & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots & \alpha_n^{k-1} \end{pmatrix}$$

avec à droite la "matrice d'évaluation" correspondant à la matrice génératrice.

Ce code a pour paramètres:

- longueur  $n$
- dimension  $q^k$ . Pour établir ceci il suffit de montrer que tout polynôme donne un mot code différent. Si il existait  $f_1 \neq f_2$  t.q.  $f_1(\alpha_i) = f_2(\alpha_i) \forall i$  et telles que  $\deg(f_1) < k$  et  $\deg(f_2) < k$ , alors en posant

$$g = f_1 - f_2$$

on aurait que le nombre de racines de  $g$  est  $\geq n \geq k$  alors que  $\deg(g) < k$  ce qui est impossible.

- une distance minimale  $d = n - k + 1$ . En effet

$$d = \min_{c \in C, c \neq 0} w(c)$$

et comme

$$w(c) = n - \text{nbre racines}$$

et que le nombre de racines est au plus  $k - 1$ , on a que

$$d \geq n - (k - 1).$$

Il suit que  $d = n - k + 1$  par la borne supérieure de Singleton.

**Observation 1** Les codes de Reed-Solomon sont donc des codes MDS.

**Observation 2**  $RS(n, k - 1) \subseteq RS(n, k)$  car les polynômes de degré  $\leq k - 1$  sont aussi de degré  $\leq k$ .

**Observation 3** Éliminer (ponctuer) une même coordonnée à tous les mots codes d'un code de  $RS(n, k)$  donne un code de Reed Solomon (on fait une évaluation en moins) pour autant que  $n - 1 \geq k$ .

## 1.1 Décodage (Berlekamp-Welch, 1986)

Soit  $C$  un code RS,  $(\alpha_1, \alpha_2, \dots, \alpha_n) \in \mathbb{F}_q^n$ , et  $c \in C$  tel que  $c_i = f^*(\alpha_i)$

On observe  $y = c + e$  et l'on veut retrouver  $y$ .

CAS 1: Pas d'erreur

$$y_i = f^*(\alpha_i) \forall i.$$

Alors

$$\begin{pmatrix} y_1 \\ \cdot \\ \cdot \\ y_n \end{pmatrix} = \begin{pmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdot & \cdot & \cdot & \alpha_1^{k-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdot & \cdot & \cdot & \alpha_2^{k-1} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \alpha_n^{k-1} \end{pmatrix} \cdot \begin{pmatrix} m_0 \\ \cdot \\ \cdot \\ m_{k-1} \end{pmatrix}$$

La matrice des alphas étant de rang plein (matrice de vandermonde) on peut retrouver le message  $m$  (la matrice est inversible a gauche).

CAS 2: erreurs

On définit

$$\Lambda(X) = \prod_{j:e_j \neq 0} (X - \alpha_j)$$

comme étant le *polynôme localisateur d'erreur*. On remarque que les racines de  $\Lambda$  donnent les localisations des erreurs. Si l'on parvient à connaître  $\Lambda$ , on élimine les  $y_i$  correspondants. Si le nombre d'erreurs est  $\leq d - 1$ , on pourra retrouver  $c$  (propriété MDS).

**Observation 4** *Le polynôme  $\Lambda$  satisfait*

$$\Lambda(\alpha_i) \cdot y_i = \Lambda(\alpha_i) \cdot f^*(\alpha_i)$$

*car si il y a erreur en  $i$ ,  $\Lambda(\alpha_i) = 0$ , et sinon,  $y_i = f^*(\alpha_i) = c_i$  la  $i$ ème coordonnée du vecteur envoyé.*

Le problème de décodage est donc

**Problème 1** *Trouver  $\Lambda(X)$  et  $f(X)$  tels que*

$$\Lambda(\alpha_i) \cdot (y_i - f(\alpha_i)) = 0 \quad \forall i \tag{1}$$

*avec  $\deg(f) \leq k - 1$  et  $\deg(\Lambda)$  minimal.*

Le difficulté est que (1) est une équation avec des termes multivariés (produits de coefficients de  $\Lambda$  et  $f$ ) ce qui rend la solution possible mais complexe à trouver.

## 1.2 Relaxation du problème

**Problème 2** *Etant donné  $y_1, y_2, \dots$ , trouver  $\Lambda(X)$  et  $h(X)$  tels que*

$$\Lambda(\alpha_i) \cdot y_i - h(\alpha_i) = 0 \quad \forall i \tag{2}$$

*avec  $\deg(h) < k + \deg(\Lambda)$  et  $\deg(\Lambda)$  minimal (on a juste remplacé le terme non linéaire  $\Lambda \cdot f$  dans (1) par un terme linéaire  $h$ ).*

Le problème s'écrit alors :

$$\begin{pmatrix} y_1 & 0 & & \\ 0 & y_2 & & \\ 0 & 0 & \cdot & \\ \cdot & \cdot & \cdot & y_n \end{pmatrix} \begin{pmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdot & \alpha_1^t \\ 1 & \alpha_2 & \alpha_2^2 & \cdot & \alpha_2^t \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \alpha_n^t \end{pmatrix} \begin{pmatrix} \Lambda_0 \\ \cdot \\ \cdot \\ \cdot \\ \Lambda_t \end{pmatrix} = \begin{pmatrix} 1 & \alpha_1 & \alpha_1^2 & \cdot & \alpha_1^{k+t-1} \\ 1 & \alpha_2 & \alpha_2^2 & \cdot & \alpha_2^{k+t-1} \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \alpha_n^{k+t-1} \end{pmatrix} \begin{pmatrix} h_0 \\ \cdot \\ \cdot \\ \cdot \\ h_{k+t-1} \end{pmatrix}$$

où  $t$  est le degré de  $\Lambda$ . On essaie de résoudre pour  $t = 0, t = 1, \dots$  jusqu'au moment où on trouve une solution pour  $\Lambda$  et  $h$ . Si  $h/\Lambda$  est un polynôme de degré  $< k$  alors l'algorithme produit  $\hat{f} = h/\Lambda$ . Sinon, il déclare une erreur.

1. Comment garantir qu'une paire  $(h, \Lambda)$  existe? Il suffit pour cela d'avoir au moins  $n$  degrés de liberté. Donc il suffit que

$$t + 1 + k + t = n$$

ce qui est impliqué par la condition

$$t = \left\lfloor \frac{d-1}{2} \right\rfloor$$

puisque  $d = n - k + 1$ . Donc l'algorithme trouve une paire  $(h, \Lambda)$  pour un

$$t \leq \left\lfloor \frac{d-1}{2} \right\rfloor.$$

De plus, une de ces paires  $(h, \Lambda)$  correspond au polynôme localisateur  $\Lambda(X) = \prod_{e_j \neq 0} (X - \alpha_j)$  et  $f^*(X)$ , si le nombre d'erreurs est  $\leq \lfloor \frac{d-1}{2} \rfloor$ .

2. Cette solution est-elle unique? Soit  $(h_1, \Lambda_1)$  et  $(h_2, \Lambda_2)$  deux solutions de (2) pour un même  $t \leq \lfloor \frac{d}{2} \rfloor - 1$ . Alors

$$h_1(\alpha_i) * \Lambda_2(\alpha_i) = \Lambda_1(\alpha_i) * y_i * \Lambda_2(\alpha_i) = \Lambda_1(\alpha_i) * h_2(\alpha_i) \quad i = 1, 2, \dots, n.$$

D'où

$$h_1(\alpha_i)/\Lambda_1(\alpha_i) = h_2(\alpha_i)/\Lambda_2(\alpha_i) \quad i = 1, 2, \dots, n$$

et comme

$$\deg(h_1/\Lambda_1) = \deg(h_2/\Lambda_2) = k - 1 < n$$

on déduit que

$$h_1/\Lambda_1 = h_2/\Lambda_2 = f^*$$

et donc que la solution trouvée est la bonne.

En combinant 1. et 2. il suit que la procédure de décodage s'arrête pour un

$$t \leq \left\lfloor \frac{d-1}{2} \right\rfloor$$

et que cette solution est correcte si le nombre d'erreurs est  $\leq \lfloor \frac{d-1}{2} \rfloor$ . De plus le décodage est de faible complexité; à chaque itération la résolution du système linéaire peut se faire avec complexité  $O(n^3)$  comme il y a au plus  $t \leq n$  itérations, la complexité totale est  $O(n^4)$ .

## 2 Codes BCH (Bose, Ray-Chaudhuri, Hocquenghem)

Vu: pour  $1 \leq k \leq n$  et  $\mathbb{F}_q$  t.q.  $n \leq q$  il existe un code  $RS(n, k, d = n - k + 1)$ .

Soit  $n = q = p^m$ , ou  $p$  est premier et  $m$  est entier. On définit le code

$$BCH_{p,m,d} \equiv RS[n, n - d + 1, d]_{p^m} \cap \mathbb{F}_p^n$$

I.e., le sous-code de RS obtenu par la restriction des composantes dans le corps de base  $\mathbb{F}_p$ . Se décode donc comme un code RS.

Paramètres:

- longueur  $n = p^m$
- distance minimale  $\geq d$

Remarque:

Ces codes permettent d'atteindre la borne de Hamming pour certaines petites valeurs de  $n$ .

### Théorème 1

$$\dim(BCH_{p,m,d}) \geq p^m - 1 - m \left\lceil \frac{(d-2)(d-1)}{p} \right\rceil$$

et donc pour tout  $m, t \geq 1$  entier  $BCH_{2,m,2t}$  est un  $[n, n - 1 - (2t - 1)(t - 1) \log_2 n, 2t]_2$  code.

Cette classe de codes est intéressante seulement si  $t = O(\sqrt{n/\log n})$  (ce qui donne un taux élevé et une distance minimale faible, sous linéaire).