Alain Couvreur (alain.couvreur@inria.fr)
Anne Canteaut (anne.canteaut@inria.fr)
Thomas Debris-Alazard (thomas.debris@inria.fr)

Codes correcteurs d'erreur et application à la cryptologie

2021 – 2022

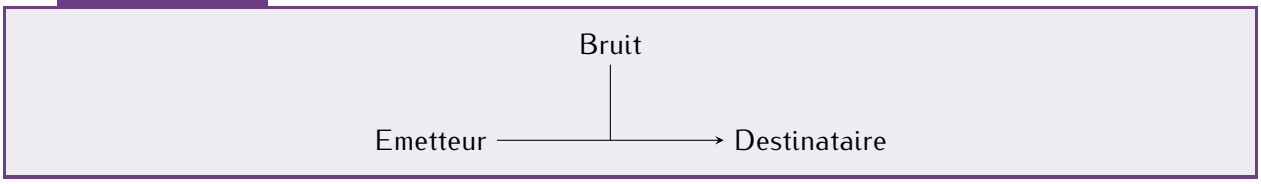
Table des matières

Première partie

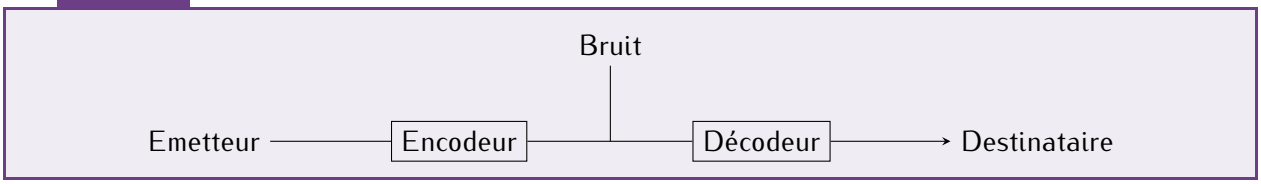
Alain Couvreur

1 Introduction à la théorie des codes

Problématique



Solution



1.1 Codes correcteurs d'erreurs

1.1.1 Premiers exemples

Le numéro de sécurité sociale contient deux numéros à la fin qui permettent de détecter s'il y a une erreur dans les premiers numéros : c'est le complément à 97 du reste de la division euclidienne du numéro principal par 97.

Exemple 1.1 **Code de répétition**

Le code le plus simple auquel on peut penser est le code de répétition :

$$\begin{array}{rccc} \text{Enc} : & \mathbb{F}_2 & \longrightarrow & \mathbb{F}_2^n \\ & b & \longmapsto & (b, ..., b) \end{array}$$

Et le décodeur est un vote majoritaire.

Mais c'est un code très peu efficace. Un code plus efficace est le code de parité, qui est également un code correcteur.

Exemple 1.2 **Code de parité (checksum)**

$$\begin{array}{rccc} \text{Enc} : & \mathbb{F}_2^{n-1} & \longrightarrow & \mathbb{F}_2^n \\ & (b_1, ..., b_{n-1}) & \longmapsto & (b_1, ..., b_{n-1}, \sum_{i=1}^{n-1} b_i) \end{array}$$

Et le décodeur vérifie si le mot reçu a un nombre pair de 1.

1.1.2 Définitions

Définition 1.3 **Code linéaire**

Un code linéaire est un sous-espace vectoriel de $\mathbb{F}_q^n$.
Ses paramètres, notés $[n, k, d]_q$ sont :

— sa longueur n

— sa dimension k

— sa distance minimale d.

Ce qui fait la richesse d'un code c'est sa métrique, qui définit la distance minimale entre ses éléments.

Définition 1.4 **Distance et poids de Hamming**

Soient $x, y \in \mathbb{F}_2^n$.
La distance de Hamming entre x et y est $d_H(x, y) := |\{i \in [\![1, n]\!] \mid x_i \neq y_i\}|$.
Le poids de Hamming de $x \in \mathbb{F}_2^n$ est $w_H(x) := d_H(x, 0)$.

On peut désormais définir la distance minimale d'un code.

Définition 1.5 **Distance minimale**

Soit $\mathcal{C} \subseteq \mathbb{F}_2^n$ un code.
La distance minimale de $\mathcal{C}$ est $d_{\min}(\mathcal{C}) := \min_{c \neq c' \in \mathcal{C}} \{d_H(c, c')\}$

Exemple 1.6

Les paramètres du code de répétition sont $[n, 1, n]$.
Les paramètres du code de parité sont $[n, n-1, 2]$.

Remarque 1.7

Si $\mathcal{C}$ est linéaire, $d_{\min}(\mathcal{C}) = \min_{c \neq 0 \in \mathcal{C}} \{w_H(c)\}$.

Pour rendre compte de l'efficacité d'un code on parle de son rendement (la quantité d'information inutile envoyée) et de sa distance relative (son potentiel de correction).

Définition 1.8 Rendement d'un code, distance relative

Soit code $\mathcal{C} \subseteq \mathbb{F}_2^n$ de paramètres $[n, k, d]$.
Son rendement est $R := \frac{k}{n}$.
Sa distance relative est $\delta := \frac{d}{n}$.

Lemme 1.9

Soit $\mathcal{C} \subseteq \mathbb{F}_2^n$ un code $[n, k, d]$.
Pour $c, c' \in \mathcal{C}$ avec $c \neq c'$,

$$B_H\left(c, \left\lfloor \frac{d-1}{2} \right\rfloor\right) \cap B_H\left(c', \left\lfloor \frac{d-1}{2} \right\rfloor\right) = \varnothing$$

1.1.3 Applications

Les codes correcteurs d'erreurs sont appliquées notamment

— dans les communications et le stockage
— sur le cloud pour reconstruire un serveur qui tombe en panne
— en cryptographie symétrique (A. Canteaut)
— en cryptographie à clé publique (T. D.-A.)
— en calcul multiparti, pour le partage de secret

Pour représenter un code il suffit d'avoir une base.

Définition 1.10 Matrice génératrice

Soit $\mathcal{C} \subseteq \mathbb{F}_2^n$ un code.
Une matrice génératrice $\mathbb{G}$ de $\mathcal{C}$ est une matrice dont les lignes engendrent $\mathcal{C}$:

$$\mathcal{C} = \left\{ m\mathbb{G} \mid m \in \mathbb{F}_2^\ell \right\}$$

avec $\ell \geqslant k$ le nombre de lignes de $\mathbb{G}$.

Définition 1.11 Matrice de parité

Une matrice de parité $\mathbb{H}$ de $\mathcal{C}$ est une matrice telle que

$$\mathcal{C} = \left\{ y \in \mathbb{F}_2^n \mid \mathbb{H} y^\intercal = 0 \right\}.$$

Étant données $\mathbb{G}$ et $\mathbb{H}$ les matrices génératrices et de parité d'un code, $\mathbb{H} \cdot \mathbb{G}^\intercal = 0$.

Exemple 1.12

Pour le code de répétition, $\mathbb{G} = (1 \ldots 1)$ et $\mathbb{H} = \begin{pmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \cdots & 0 \\ & \ddots & \ddots & & \\ 0 & \cdots & 0 & 1 & 1 \end{pmatrix}$.

Pour le code de parité on inverse les deux.

1.1.4 Code de Hamming

Définition 1.13 Code de Hamming

Le code de Hamming est le code $\mathcal{C}$ de matrice de parité

$$\mathbb{H} = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{pmatrix}.$$

Propriété 1.14

Le code de Hamming est un code $[7, 4, 3]$.

Preuve.
$\dim(\mathcal{C}) = 4$ car $\mathcal{C} = \ker(\mathbb{H})$ et $\mathrm{rk}\mathbb{H} = 3$.

Lemme 1.15

La distance minimale d'un code de matrice de parité $\mathbb{H}$ est le nombre minimal de colonnes de $\mathbb{H}$ linéairement liées.

Preuve en exercice et dans les notes officielles.

La distance minimale du code de Hamming est 3 car :

— C'est > 1 car toutes les colonnes sont non nulles.

— C'est > 2 car toutes les colonnes sont deux à deux distinctes.

— C'est 3 car $\begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 0 & 0 \end{pmatrix} \in \mathcal{C}$.

Remarque 1.16

Si on prend un code de répétition de longueur 7 : $[7, 1, 7]$, on peut corriger jusque 3 erreurs.

	Répétition	Hamming
Taux de correction	$\frac{3}{7}$	$\frac{1}{7}$
Rendement	$\frac{1}{7}$	$\frac{4}{7}$

2 | Décodage et théorème de Shannon

2.1 | Décodeurs

Définition 2.1 Décodeur

Soit $\mathcal{C} \subseteq \mathbb{F}_2^n$ un code.
Un décodeur pour $\mathcal{C}$ est une fonction

$$\mathcal{D} : \mathbb{F}_2^n \longrightarrow \mathcal{C} \cup \{?\}$$

tel que pour tout $c \in \mathcal{C}$, $\mathcal{D}(c) = c$.

Exemple 2.2 Maximum likelihood decoder

$$\mathcal{D}_{MV} : \begin{array}{ccc} \mathbb{F}_2^n & \longrightarrow & \mathcal{C} \cup \{?\} \\ y & \longmapsto & \begin{cases} c \in \mathcal{C} \text{ tq } d_H(y,c) = \min_{c' \in \mathcal{C}} d_h(y,c') \text{ s'il est unique} \\ ? \text{ sinon} \end{cases} \end{array}$$

Dans la littérature, on distingue deux types d'algorithmes de décodage :

1. Les décodeurs combinatoires, qui corrigent tout motif d'erreur de poids inférieur à un certain seuil

t. Notons que
— soit $t \leqslant \left\lfloor \frac{d-1}{2} \right\rfloor$
— soit $t > \left\lfloor \frac{d-1}{2} \right\rfloor$, auquel cas on généralise aux décodeurs en liste $\mathcal{D} : \mathbb{F}_2^n \to \mathcal{P}(\mathcal{C})$.

2. Les décodeurs probabilistes, qui peuvent échouer avec une certaine probabilité sur une instance aléatoire.

2.2 Canaux

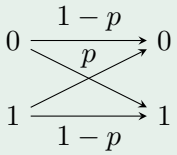
2.2.1 Canaux

Un canal est un modèle probabiliste pour modéliser les erreurs. Dans ce cours on va se focaliser sur le cas de canaux sans mémoire.

Définition 2.3 — Canal sans mémoire

Un canal sans mémore est une suite de variables aléatoires indépendantes $(e_1, ..., e_n)$ à valeurs dans $\mathbb{F}_2$.

Exemple 2.4 — Canal binaire symétrique de paramètre $p < \frac{1}{2}$ (BSC)



Les e_i suivent une loi de Bernoulli $\mathcal{B}(p)$.

2.2.2 Décodage "probabiliste"

On se donne $\mathcal{C} \subseteq \mathbb{F}_2^n$ et $\mathcal{D} : \mathbb{F}_2^n \to \mathcal{C} \cup \{?\}$.

$$\mathbb{P}_{\text{échec}}(\mathcal{C}, \mathcal{D}) := \mathop{\mathbb{P}}_{\substack{c \sim \mathcal{U}(\mathcal{C}) \\ e \sim \mathrm{BSC}(p)}} (\mathcal{D}(c+e) \neq c)$$

On veut prouver l'existence de couples $(\mathcal{C}, \mathcal{D})$ pour lesquels $\mathbb{P}_{\text{échec}}(\mathcal{C}, \mathcal{D})$ est aussi petite que possible.

Question

Existe-t-il des suites de couples $(\mathcal{C}_s, \mathcal{D}_s)$ telles que $\mathcal{C}_s \subseteq \mathbb{F}_2^{n_s}$, $n_s \to \infty$ et $\mathbb{P}_{\text{échec}}(\mathcal{C}_s, \mathcal{D}_s) \underset{s\to\infty}{\longrightarrow} 0$?

La réponse est oui. Par exemple avec $\mathcal{C}_s$ le code de répétition de $\mathbb{F}_2^s$ et $\mathcal{D}_s$ le décodeur par vote majoritaire ($\mathcal{D}_{MV}$).

Preuve.
On note $y := c + e$ avec $c \sim \mathcal{U}(\mathcal{C})$ et $e \sim \mathrm{BSC}(p)$.

$$\mathcal{D}_{MV} = \begin{cases} (0...0) & \text{si } w_H(y) < \frac{s}{2} \\ (1...1) & \text{si } w_H(y) > \frac{s}{2} \\ ? & \text{si } w_H(y) = \frac{s}{2} \end{cases}$$

donc

$$\mathbb{P}_{\text{échec}}(\mathcal{C}_s, \mathcal{D}_s) = \mathbb{P}_{e \sim \mathrm{BSC}(p)}(w_H(e) \geqslant \frac{s}{2}).$$

$e = (e_1, ..., e_s)$ où les e_i sont indépendantes de loi $\mathcal{B}(p)$ et $p < \frac{1}{2}$.
$\mathbb{E}(w_H(e)) = ps$ et par indépendance $\mathbb{V}(e) = p(1-p)s$.
Soit $\varepsilon \in]p, \frac{1}{2}[$. Alors par Pafnouti,

$$\mathbb{P}_{\text{échec}}(\mathcal{C}_s, \mathcal{D}_s) = \mathbb{P}_{e \sim \mathrm{BSC}(p)}(w_H(e) \geqslant \frac{s}{2})$$

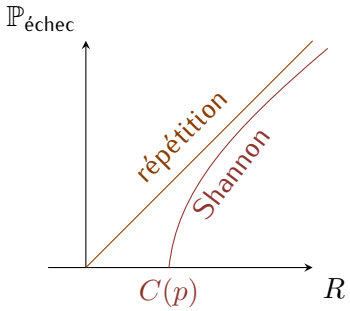
$$
\begin{aligned}
&\leqslant \mathbb{P}_{e\sim \mathrm{BSC}(p)}(w_H(e) \geqslant (p+\varepsilon)s) \\
&\leqslant \mathbb{P}_{e\sim \mathrm{BSC}(p)}(w_H(e) - ps \geqslant \varepsilon s) \\
&\leqslant \mathbb{P}_{e\sim \mathrm{BSC}(p)}(|w_H(e) - ps| \geqslant \varepsilon s) \\
&\leqslant \mathbb{P}_{e\sim \mathrm{BSC}(p)}(|w_H(e) - \mathbb{E}(w_H(e)| \geqslant \varepsilon s) \\
&\leqslant \frac{\mathbb{V}(w_H(e))}{\varepsilon^2 s^2} \\
&\leqslant O(\frac{1}{s})
\end{aligned}
$$

2.3 Théorème de Shannon

Question

Existe-t-il des suites de couples $(\mathcal{C}_s, \mathcal{D}_s)$ telles que $\mathcal{C}_s \subseteq \mathbb{F}_2^{n_s}$, $n_s \to \infty$ et $\mathbb{P}_{\text{échec}}(\mathcal{C}_s, \mathcal{D}_s) \underset{s\to\infty}{\longrightarrow} 0$, mais aussi $R_s := \frac{\dim \mathcal{C}_s}{n_s} \to R > 0$?

La réponse est encore oui.

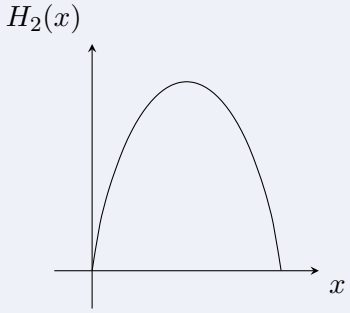


2.3.1 Entropie et boules de Hamming

Définition 2.5 Fonction d'entropie binaire

On définit la fonction d'entropie binaire par

$$
H_2 : \begin{array}{ccc} [0,1] & \longrightarrow & \mathbb{R} \\ x & \longmapsto & \begin{cases} -x\log_2 x - (1-x)\log_2(1-x) & \text{si } 0 < x < 1 \\ 0 & \text{sinon} \end{cases} \end{array}
$$



Proposition 2.6

Soit $0 < p < \frac{1}{2}$.
Soit $V(r,n)$ le cardinal volume de la boule de Hamming $B_H(0,r) \subseteq \mathbb{F}_2^n$, $V(r,n) := \sum_{i=0}^{r} \binom{r}{i}$.
On a

1. $V(pn,n) \leqslant 2^{nH_2(p)}$
2. $\forall \varepsilon > 0, \exists N \geqslant 0, \forall n \geqslant N, 2^{n(H(p)-\varepsilon)} \leqslant V(pn,n)$

2.3.2 | Théorème de Shannon

Théorème 2.7 **Théorème de Shannon (1949)**

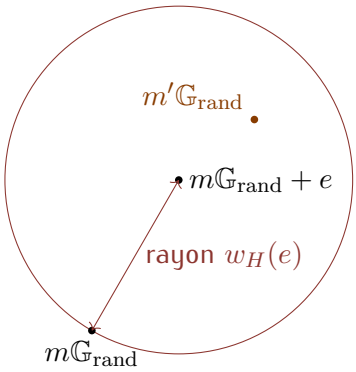
Soient $0 < p < \frac{1}{2}$ et $0 < \varepsilon < \frac{1}{2} - p$. On a les propriétés suivantes.

1. $\exists \delta > 0, \exists N > 0, \forall n \geqslant N, \exists (\mathcal{C}, \mathcal{D}), \mathcal{C} \subseteq \mathbb{F}_2^n$ de dimension $k = \lfloor (1 - H(p) - \varepsilon)n \rfloor$ et $\mathbb{P}_{\text{échec}}(\mathcal{C}, \mathcal{D}) < 2^{-\delta n}$
2. $\exists N > 0, \forall n \geqslant N$, pour tout couple $(\mathcal{C}, \mathcal{D})$ avec $\mathcal{C} \subseteq \mathbb{F}_2^n$ et $\dim(\mathcal{C}) = \lfloor (1 - H_2(p) + \varepsilon)n \rfloor$ alors $\mathbb{P}_{\text{échec}}(\mathcal{C}, \mathcal{D}) \geqslant \frac{1}{2}$

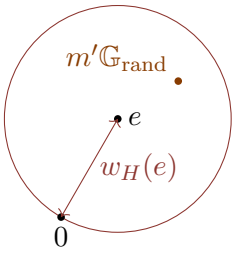
La quantité $C(p) = 1 - H(p)$ est appelée capacité du canal binaire symétrique.

Preuve (fausse).

On prend pour $\mathcal{D}$ le décodeur $\mathcal{D}_{MV}$. On se donne un code aléatoire $\mathcal{C} \in \mathbb{F}_2^n$ avec n assez grand, i.e. $\mathbb{G}_{\text{rand}} \in \mathbb{F}_2^{k \times n}$.
$\mathcal{C} := m\mathbb{G}_{\text{rand}} \mid m \in \mathbb{F}_2^k$
$k = \lfloor (1 - H(p) - \varepsilon)n \rfloor$
$\mathbb{P}_{\text{échec}}(\mathcal{C}_{\text{rand}}, \mathcal{D}_{MV}) = \mathbb{P}(\mathcal{D}_{MV}(m\mathbb{G}_{\text{rand}} + e) \neq m\mathbb{G}_{\text{rand}})$ où $m \sim \mathcal{U}(\mathbb{F}_2^k)$ et $e \sim \text{BSC}(p)$.
Le cas d'échec est le suivant :



En fait l'échec ne dépend que de e puisqu'on peut translater :



On a

$$\mathbb{P}_{\text{échec}}(\mathcal{C}_{\text{rand}}, \mathcal{D}_{MV}) = \mathbb{P}(\exists m \in \mathbb{F}_2^k \setminus \{0\} \mid n\mathbb{G}_{\text{rand}} \in B_H(e, w_H(e)))$$

Donc par mensonge,

$$\begin{aligned} \mathbb{P}_{\text{échec}}(\mathcal{C}_{\text{rand}}, \mathcal{D}_{MV}) &= \mathbb{P}(\exists m \in \mathbb{F}_2^k \setminus \{0\} \mid n\mathbb{G}_{\text{rand}} \in B_H(e, pn)) \\ &\leqslant \sum_{m \in \mathbb{F}_2^k \setminus \{0\}} \mathbb{P}(m\mathbb{G}_{\text{rand}} \in B(e, pn) \end{aligned}$$

$m\mathbb{G}_{\text{rand}} \sim \mathcal{U}(\mathbb{F}_2^k \setminus \{0\})$ donc

$$\begin{aligned} \mathbb{P}_{\text{échec}}(\mathcal{C}_{\text{rand}}, \mathcal{D}_{MV}) &\leqslant \sum_{m \in \mathbb{F}_2^k \setminus \{0\}} \frac{\text{Vol}(pn, n) - 1}{2^k - 1} \\ &\leqslant \sum_{m \in \mathbb{F}_2^k \setminus \{0\}} \frac{2^{nH(p)}}{2^n} \\ &\leqslant 2^{k + nH(p) - n} \\ &\leqslant 2^{-n\varepsilon} \end{aligned}$$

3 | Borne sur les paramètres des codes

Ici "borne sup" signifie : "il existe une fonction $\Phi : \mathbb{N}^3 \to \mathbb{R}$ telle que pour tout code $\mathcal{C}$ de paramètres $[n,k,d]_q$, on a $\Phi(n,k,d) \leqslant 0$."
"Borne inf" signifie : "il existe une fonction $\Psi : \mathbb{N}^3 \to \mathbb{R}$ telle qu'il existe un code $\mathcal{C}$ de paramètres $[n,k,d]_q$ vérifiant $\Psi(n,k,d) \geqslant 0$."

3.1 | Bornes supérieures

3.1.1 | La borne de Singleton

Théorème 3.1 — Borne de Singleton

Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ de paramètres $[n,k,d]_q$.
Alors $k + d \leqslant n + 1$.

Preuve.
Soit $G \in \mathbb{F}_q^{k\times n}$ une matrice génératrice de $\mathcal{C}$.

Remarque

Effectuer des opérations élémentaires sur les *lignes* ne change pas le code.

On met la matrice G sous forme échelonnée, par élimination gaussienne. On obtient G' :

$$G' = \left(\begin{array}{ccc|c} * & & & \\ 0 & * & (*) & \\ \vdots & \ddots & & (*) \\ \vdots & (0) & \ddots & \\ \underbrace{0 \quad \ldots \quad 0}_{k-1} & & & \end{array}\right)$$

La dernière ligne de G' est de poids $\leqslant n-k+1$ donc il existe $c \in \mathcal{C} \setminus \{0\}, w(c) \leqslant n-k+1$.

Ce théorème est également vrai pour les codes non linéaires.

Théorème 3.2 — Borne de Singleton (code non linéaire)

Soit $\mathcal{C}_{\text{nl}} \subseteq \mathbb{F}_q^n$ un code non linéaire de distance minimale d.
Alors $M \leqslant q^{n-k+1}$.

Preuve.
Soit $\pi : \begin{array}{ccc} \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^{n-k+1} \\ (x_1,\ldots,x_n) & \longmapsto & (x_d,\ldots,x_n) \end{array}$.
La restriction de π à $\mathcal{C}_{\text{rnl}}$ est injective. En effet pour $c, c' \in \mathcal{C}_{\text{nl}}$,

$$\begin{aligned} \pi(c) = \pi(c') &\Rightarrow c_d = c'_d, \ldots, c_n = c'_n \\ &\Rightarrow d(c,c') \leqslant d-1 \\ &\Rightarrow c = c' \end{aligned}$$

Par conséquent : $|\mathbb{F}_q^{n-d+1}| = q^{n-d+1} \geqslant |\mathcal{C}_{\text{nl}}|$.

Question

La borne de Singleton est-elle fine ?

1. Il existe des codes dits MDS (Maximum Distance Separable) qui atteignent cette borne (ex : les codes de Reed-Solomon).
2. Les codes MDS sont courts.

Conjecture 3.3 **Conjecture MDS**

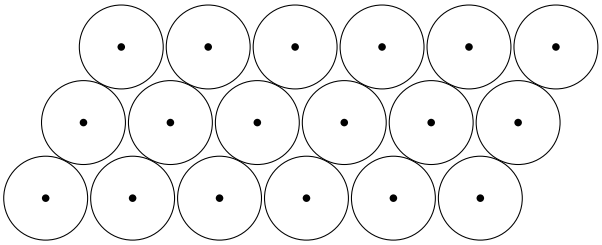
Pour tout code $\mathcal{C} \subseteq \mathbb{F}_q^n$, si $\mathcal{C}$ est MDS alors $n \leqslant q+1$. Sauf pour
— le code nul $\mathcal{C} = \{0\}$
— le code complet $\mathcal{C} = \mathbb{F}_q^n$
— les codes de répétition et de parité
— deux codes exceptionnels de dimension 3 et $q-1$ dans $\mathbb{F}_q^{q+2}$ pour $q = 2^r$.

Quoi qu'il en soit, un code MDS vérifie toujours $n \leqslant q+k-1$ (cf. notes de cours).

Asymptotiquement, pour q fixé, Singleton donne $R + \delta \leqslant 1$.

3.1.2 La borne de Hamming (Sphere packing)

L'idée est, étant donné $\mathcal{C} \subseteq \mathbb{F}_q^n$ de paramètres $[n,k,d]_q$, les boules de rayon $\lfloor \frac{d-1}{2} \rfloor$ centrées en des mots de code sont 2 à 2 disjoints.



Théorème 3.4 **Borne de Hamming**

Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ de paramètres $[n,k,d]$.
Alors le nombre de boules multiplié par leur volume est inférieur au volume total : $q^k \mathrm{Vol}_q(\lfloor \frac{d-1}{2} \rfloor, n) \leqslant q^n$.

Asymptotiquement, $R + H(\frac{\delta}{2}) \leqslant 1$.

3.1.3 La borne de Plotkin

L'idée est d'énumérer tous les mots du code sauf 0, et de les mettre dans un tableau.

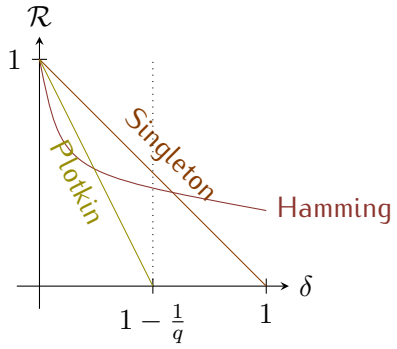
c_1			$\cdots$	
c_2			$\cdots$	
$\vdots$			$\cdots$	

Soit $\mathcal{D} := \sum_{c \in \mathcal{C} \setminus \{0\}} w(c)$ le nombre de coefficients non nuls du tableau.
Par ligne : $\mathcal{D} \geqslant d(q^k - 1)$
Par colonne : Soit $\psi_i : \begin{array}{ccc} \mathcal{C} & \longrightarrow & \mathbb{F}_q \\ c & \longmapsto & c_i \end{array}$. C'est une forme linéaire qui est soit nulle, auquel cas $\mathcal{C} = \ker \psi_i$, soit surjective, auquel cas $\dim \ker \psi_i = k-1$. Par conséquent, le nombre de 0 dans une colonne est $\geqslant q^{k-1} - 1$ (on a enlevé 0).
Donc le nombre de coefficients non nuls dans une colonne est $\leqslant (q^k - 1) - (q^{k-1} - 1) \leqslant q^k - q^{k-1}$.
D'où $\mathcal{D} \leqslant n(q^k - q^{k-1})$.

Théorème 3.5 Borne de Plotkin

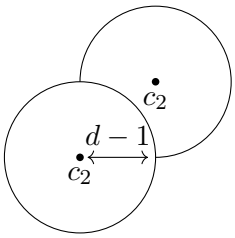
Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ un code $[n,k,d]$.
Alors $d \leqslant n\frac{q^k - q^{k-1}}{q^k - 1}$.

Asymptotiquement, $\frac{d}{n} \leqslant \frac{1-1/q}{1-1/q^k}$.
Avec $d = \delta n$ et $k = \mathcal{R}n$, en faisant tendre n vers ∞, $\delta \leqslant 1 - \frac{1}{q}$.
On admet $\mathcal{R} \leqslant \max(1, \frac{q}{q-1}\delta, 0)$.



3.2 | Une borne inférieure : la borne de Gilbert-Varshamov

L'idée est de se fixer d, et ensuite construire un code non linéaire le plus grand possible.



Théorème 3.6

Il existe un code non linéaire $\mathcal{C} \subseteq \mathbb{F}_q^n$ de distance minimale d tel que

$$|\mathcal{C}|\mathrm{Vol}_q(d-1,n) \geqslant q^n.$$

Preuve.
Supposons que pour tout $\mathcal{C} \subseteq \mathbb{F}_q^n$ de dimension d on ait $|\mathcal{C}|\mathrm{Vol}_q(d-1,n) < q^n$.
Soit $\mathcal{C}$ un tel code de cardinal maximal. Alors

$$\left|\bigcup_{c\in\mathcal{C}} B(c,d-1)\right| \leqslant |\mathcal{C}|\mathrm{Vol}_q(d-1,n) < q^n$$

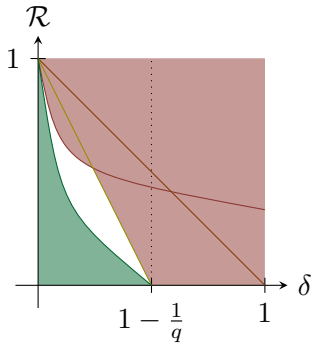
Donc $\mathbb{F}_q^n \setminus \bigcup_{c\in\mathcal{C}} B(c,d-1) \neq \varnothing$.
Soit $c_0 \in \mathbb{F}_q^n \setminus \bigcup_{c\in\mathcal{C}} B(c,d-1)$. Alors $\mathcal{C} \cup \{c_0\}$ est de distance minimale d. Ce qui contredit la maximalité de $\mathcal{C}$.

Asymptotiquement, il existe une suite de codes $(\mathcal{C}_s)_{s\in\mathbb{N}}$ de paramètres $[n_s,k_s,d_s]$ tels que $n_s \underset{s\to\infty}{\longrightarrow} \infty$, $\frac{d_s}{n_s} \underset{s\to\infty}{\longrightarrow} \delta$, $\frac{k_s}{n_s} \underset{s\to\infty}{\longrightarrow} \mathcal{R}$ et $\mathcal{R} \geqslant 1 - H_q(s)$.

Définition 3.7

$$H_q(x) = \begin{cases} 0 & \text{si } x = 0 \text{ ou } 1 \\ x\log_q(q-1) - x\log_q x - (1-x)\log_q(1-x) & \text{sinon} \end{cases}.$$

On admet que la borne reste vraie si on se restreint aux codes linéaires : il existe des codes tels que $q^{k_s}\mathrm{Vol}_q(d_s-1,n) \geqslant q^{n_s}$, donc $q^{k_s}q^{n_sH_q(\delta_s-\frac{1}{n_s})} \geqslant q^{n_s}$. Donc en passant au $\log_q$, $k_s+n_sH_q(\delta_s-\frac{1}{n_s} \geqslant n_s$, donc $\mathcal{R}_s \geqslant 1-H_q(\delta_s-\frac{1}{n_s})$, soit asymptotiquement $\mathcal{R} \geqslant 1-H_q(\delta)$.



Théorème 3.8

Soient $0 < \delta \leqslant 1-\frac{1}{q}$ et $\varepsilon > 0$. Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ un code aléatoire de dimension $k = (1-H_q(\delta)-\varepsilon)n$. Alors $\mathbb{P}(d_{\min}(\mathcal{C}) < \delta n) = O(q^{-\varepsilon n})$.

Preuve (esquisse).
Soit $\mathbb{G} \in \mathbb{F}_q^{k\times n}$ aléatoire uniforme (parmi les matrices de rang k). Soit $\mathcal{C} = \{m\mathbb{G} \mid m \in \mathbb{F}_q^k\}$.

Lemme 3.9

Soit $m \in \mathbb{F}_q^k \setminus \{0\}$. Alors $m\mathbb{G}$ est une variable aléatoire de loi uniforme sur $\mathbb{F}_q^n \setminus \{0\}$.

Lemme 3.10

Soit x un vecteur aléatoire uniforme dans $\mathbb{F}_q^n$. Alors $\mathbb{P}(w_H(x) < \delta n) = \dfrac{\mathrm{Vol}(\delta n-1,n)}{q^n}$.

Alors $\mathbb{P}(w_H(x) < \delta n) \leqslant \dfrac{q^{nH_q(\delta)}}{q^n} \leqslant q^{n(H_q(\delta)-1)}$.

$$\begin{aligned}
\mathbb{P}(d_{\min}(\mathcal{C}) \leqslant \delta n) &= \mathbb{P}(\exists m \in \mathbb{F}_q^k \setminus \{0\} \mid w_H(m\mathbb{G}) < \delta n) \\
&= \mathbb{P}\Big(\bigcup_{m\in\mathbb{F}_q^k\setminus\{0\}} \{w_H(m\mathbb{G}) < \delta n\}\Big) \\
&\leqslant \sum_{m\in\mathbb{F}_q^k\setminus\{0\}} \mathbb{P}(w_H(m\mathbb{G}) < \delta n) \\
&\leqslant \sum_{m\in\mathbb{F}_q^k\setminus\{0\}} q^{n(H_q(\delta)-1)} \\
\mathbb{P}(d_{\min}(\mathcal{C}) < \delta n) &\leqslant q^k q^{n(H_q(\delta)-1)} \\
&\leqslant q^{n(1-H(\varepsilon)-\varepsilon)+n(H_q(\delta)-1)} \\
&\leqslant q^{-n\varepsilon}
\end{aligned}$$

Théorème 3.11

Soient $0 < \delta < 1-\frac{1}{q}$ et $\varepsilon > 0$. Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ un code aléatoire de dimension $k = (1-H_q(\delta)+\varepsilon)n$. Alors $\mathbb{P}(d_{\min}(\mathcal{C}) > \delta n) = O(q^{-n\varepsilon})$.

4 | Dualité

On munit $\mathbb{F}_q^n$ de la forme bilinéaire symétrique

$$\langle \cdot, \cdot \rangle : \begin{array}{ccc} \mathbb{F}_q^n \times \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q \\ (x, y) & \longmapsto & \sum_{i=1}^n x_i y_i \end{array}$$

Attention

$\langle \cdot, \cdot \rangle$ n'est pas un produit scalaire.
En particulier $\langle x, x \rangle = 0 \not\Rightarrow x = 0$.

Définition 4.1 **Dual d'un code**

Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ un code.
On définit son code dual par

$$\mathcal{C}^\perp := \{x \in \mathbb{F}_q^n \mid \forall c \in \mathcal{C}, \langle x, c \rangle = 0\}.$$

Il est possible que $\mathcal{C} \cap \mathcal{C}^\perp \neq \{0\}$. Il est même possible que $\mathcal{C} \subseteq \mathcal{C}^\perp$.

4.1 Propriétés

Proposition 4.2

Soient $\mathbb{G}$ et H des matrices respectivement génératrice et de parité d'un code $\mathcal{C} \subseteq \mathbb{F}_q^n$.
Alors $\mathbb{G}$ est matrice de parité de $\mathcal{C}^\perp$ et H est matrice génératrice de $\mathcal{C}^\perp$.

Preuve.
Montrons que H est génératrice de $\mathcal{C}^\perp$.

Soit $\mathcal{C}_0$ le code de matrice génératrice H.
Toute ligne de H est orthogonale à tout $c \in \mathcal{C}$: $\mathcal{C} = \{x \in \mathbb{F}_q^n \mid Hx^\top = 0\}$. Donc $\mathcal{C}_0 \subseteq \mathcal{C}^\perp$.
En particulier, $\dim \mathcal{C}^\perp \geqslant \dim \mathcal{C}_0 = n - k$.

Toute ligne de $\mathbb{G}$ est orthogonale à tout $c \in \mathcal{C}^\perp$ (les lignes de $\mathbb{G}$ sont dans $\mathcal{C}$) : $\mathcal{C}^\perp \subseteq \{x \in \mathbb{F}_q^n \mid \mathbb{G}x^\top = 0\}$ (le noyau à droite de $\mathbb{G}$).
Donc $\dim \mathcal{C}^\perp \leqslant n - \mathrm{rk}\mathbb{G} = n - k$. D'où $\dim \mathcal{C}^\perp = n - k$ et $\mathcal{C}^\perp = \mathcal{C}_0$.

Proposition 4.3

Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ un code de dimension k.
Alors

1. $\dim \mathcal{C}^\perp = n - k$
2. $(\mathcal{C}^\perp)^\perp = \mathcal{C}$
3. $\mathcal{C} \subseteq \mathcal{D} \Longrightarrow \mathcal{D}^\perp \subseteq \mathcal{C}^\perp$
4. $(\mathcal{C} + \mathcal{D})^\perp = \mathcal{C}^\perp \cap \mathcal{D}^\perp$
5. $(\mathcal{C} \cap \mathcal{D})^\perp = \mathcal{C}^\perp + \mathcal{D}^\perp$

Fait 4.4

Le code de répétition et le code de parité sont duaux.

4.2 Relations métriques, théorème de McWilliams

Question

Y a-t-il une formule donnant $d_{\min}(\mathcal{C})$ et $d_{\min}(\mathcal{C}^\perp)$?

La réponse est non ! On peut prouver l'existence de suites de codes $(\mathcal{C}_s)_{s\in\mathbb{N}}$ de paramètres $[n_s, k_s, d_s]$ tels que $n_s \to \infty$, $k_s = \Omega(n_s)$, $d_s = \Omega(n_s)$, $\dim \mathcal{C}_s^\perp = \Omega(n_s)$, $d_{\min}(\mathcal{C}_s^\perp) = \Omega(n_s)$, mais aussi des suites $(\mathcal{C}_s)$ tels que $\dim \mathcal{C}_s = \Omega(n_s)$, $d_{\min}(\mathcal{C}_s) = \Omega(n_s)$, $\dim \mathcal{C}_s^\perp = \Omega(n_s)$, $d_{\min}(\mathcal{C}_s^\perp) = O(1)$ (codes LDPC).
La distance minimale n'est peut-être alors pas assez informative.

Définition 4.5 — **Énumérateur des poids**

Soient $\mathcal{C} \subseteq \mathbb{F}_q^n$ et $t \in [\![0, n]\!]$.
On pose $w_t(\mathcal{C}) = |\{c \in \mathcal{C} \mid w_h(c) = t\}|$.
On définit le polyôme $\mathcal{P}_{\mathcal{C}}(x, y) = \sum_{i=0}^{n} w_i(\mathcal{C}) x^i y^{n-i}$.

Théorème 4.6 — **Théorème de McWiliams**

Soit $\mathcal{C} \subseteq \mathbb{F}_2^n$ [a].
Alors $\mathcal{P}_{\mathcal{C}^\perp}(x, y) = \frac{1}{|\mathcal{C}|}\mathcal{P}_{\mathcal{C}}(y - x, y + x)$.

a. Voir le poly officiel pour le cas $\mathbb{F}_q^n$.

Preuve.

Lemme 4.7

Soit $f : \mathbb{F}_2^n \to \mathbb{C}$. Soit $\widehat{f} : \mathbb{F}_2^n \to \mathbb{C}$ définie par

$$\forall u \in \mathbb{F}_2^n, \widehat{f}(u) = \sum_{v\in\mathbb{F}_2^n} (-1)^{\langle u,v\rangle} f(v).$$

Alors pour $\mathcal{C} \subseteq \mathbb{F}_2^n$ un code,

$$\sum_{u\in\mathcal{C}^\perp} f(u) = \frac{1}{|\mathcal{C}|} \sum_{v\in\mathcal{C}} \widehat{f}(v).$$

Preuve.

$$\begin{aligned}\sum_{v\in\mathcal{C}} \widehat{f}(v) &= \sum_{v\in\mathcal{C}} \sum_{u\in\mathbb{F}_2^n} (-1)^{\langle u,v\rangle} f(u) \\ &= \sum_{v\in\mathcal{C}} f(u) \sum_{u\in\mathbb{F}_2^n} (-1)^{\langle u,v\rangle}\end{aligned}$$

Or

$$\sum_{v\in\mathcal{C}} (-1)^{\langle u,v\rangle} = \begin{cases} |\mathcal{C}| & \text{si } u \in \mathcal{C}^\perp \\ 0 & \text{si } u \notin \mathcal{C}^\perp \end{cases}$$

En effet, si $u \notin \mathcal{C}^\perp$ alors l'application $\psi_u : \begin{array}{ccc} \mathcal{C} & \longrightarrow & \mathbb{F}_2 \\ v & \longmapsto & \langle u, v\rangle \end{array}$ est une forme linéaire non nulle.
Son noyau est un hyperplan de $\mathcal{C}$, i.e. un sous-espace de dimension $\dim \mathcal{C}$.
Donc $|\ker \psi_u| = |\{v \in \mathcal{C} \mid \langle u, v\rangle = 0\}| = 2^{\dim \mathcal{C}-1}$.
D'où

$$\begin{aligned}\sum_{v\in\mathcal{C}} (-1)^{\langle u,v\rangle} &= \sum_{v\in\ker\psi_u} 1 + \sum_{v\in\mathcal{C}\setminus\ker\psi_u} (-1) \\ &= 2^{\dim\mathcal{C}-1} - 2^{\dim\mathcal{C}-1} \\ &= 0\end{aligned}$$

On fixe $x, y \in \mathbb{C}^*$.
On va appliquer le lemme à la fonction $f : \begin{array}{ccc} \mathbb{F}_2^n & \longrightarrow & \mathbb{C} \\ u & \longmapsto & x^{n-w_H(u)} y^{w_H(u)} \end{array}$.

$$\begin{aligned}
\widehat{f}(v) &= \sum_{u\in\mathbb{F}_2^n} (-1)^{\langle u,v\rangle} x^{n-w(u)} y^{w(u)} \\
&= \sum_{u\in\mathbb{F}_2^n} (-1)^{u_1v_1}\dots(-1)^{u_nv_n} x^{1-u_1}\dots x^{1-u_n} y^{u_1}\dots y^{v_n} \\
&= \sum_{(u_1,\dots,u_n)\in\mathbb{F}_2^n} \prod_{i=1}^{n} (-1)^{u_iv_i} x^{1-u_i} y^{u_i} \\
&= \prod_{i=1}^{n} \left(\sum_{t\in\mathbb{F}_2} (-1)^{v_it} x^{1-t} y^t \right) \\
&= \prod_{i=1}^{n} (x + (-1)^{v_i} y) \\
&= (x+y)^{n-w_H(v)} (x-y)^{w_H(v)}
\end{aligned}$$

Mainteant on relie ça aux polynômes $\mathcal{P}_{\mathcal{C}}$ et $\mathcal{P}_{\mathcal{C}^\perp}$:

$$\mathcal{P}_{\mathcal{C}}(x,y) = \sum_{u\in\mathcal{C}} x^{w_H(u)} y^{n-w_H(u)}$$

$$\begin{aligned}
\mathcal{P}_{\mathcal{C}^\perp}(x,y) &= \sum_{u\in\mathcal{C}^\perp} \underbrace{x^{w_H(u)} y^{n-w_H(u)}}_{f(u)} \\
&= \frac{1}{|\mathcal{C}|} \sum_{v\in\mathcal{C}} \underbrace{(x-y)^{w_H(v)} (x+y)^{n-w_H(v)}}_{\widehat{f}(v)} \\
&= \frac{1}{|\mathcal{C}|} \mathcal{P}_{\mathcal{C}}(x-y, x+y).
\end{aligned}$$

On a donc montré que pour tous $(x,y) \in \mathbb{C}^{*2}$ on a

$$\mathcal{P}_{\mathcal{C}^\perp}(x,y) = \frac{1}{|\mathcal{C}|} \mathcal{P}_{\mathcal{C}}(x-y, x+y).$$

Par le théorème de prolongement des identités algébriques, les polynômes sont donc égaux :

$$\mathcal{P}_{\mathcal{C}^\perp}(X,Y) = \frac{1}{|\mathcal{C}|} \mathcal{P}_{\mathcal{C}}(X-Y, X+Y).$$

5 | Codes de Reed-Solomon

5.1 | Définition

Définition 5.1 **Code de Reed-Solomon**

Soient $x_1, \dots, x_n \in \mathbb{F}_q$ deux à deux distincts.
On définit le code de Reed-Solomon de dimension k et de support $x = (x_1, \dots, x_n)$ par

$$\mathcal{RS}_k(x) := \{(f(x_1), \dots, f(x_n)) \mid f \in \mathbb{F}_q[X], \deg f < k\}.$$

Propriété 5.2

Le code $\mathcal{RS}_k(x)$ a pour paramètres $[n, k, n-k+1]_q$.

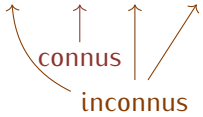
Preuve.
Considérons l'application d'encodage $\mathrm{ev}_x \left\{ \begin{array}{ccc} \mathbb{F}_q[X]_{<k} & \longrightarrow & \mathbb{F}_q^n \\ f & \longmapsto & (f(x_1),...,f(x_n)) \end{array} \right.$.
Soit $f \in \mathbb{F}_q[X]_{<k} \setminus \{0\}$. Supposons que $w_H(\mathrm{ev}_x(f)) < n-k+1$, alors il existe strictement plus de $k-1$ éléments x_i parmi $x_1,...,x_n$ tels que $f(x_i)=0$. Or les x_i sont distnicts, donc f a strictement plus de $k-1$ racines, donc $f=0$. Diantre ! Vetruchou ! L'aventure est finie.
Par conséquent

1. Pour tout $c \in \mathcal{RS}_k(x) \setminus \{0\}$, $w_H(c) \geqslant n-k+1$, donc $d_{\min}(\mathcal{RS}_k(x)) \geqslant n-k+1$.
2. Pour tout $f \in \mathbb{F}_q[X]_{<k} \setminus \{0\}$, $\mathrm{ev}_x(f) \neq 0$ donc ev_x est injective, donc $\dim \mathcal{RS}_x(x) = \dim \mathbb{F}_q[X]_{<k} = k$.
3. Par Singleton, on a donc $d = n-k+1$.

5.2 | L'algorithme de décodage de Welch-Berlekamp

Soient $x=(x_1,...,x_n)$, $c=\mathrm{ev}_x(f) \in \mathcal{RS}_k(x)$, $e \in \mathbb{F}_q^n$ et $t=w_H(e)$.
On suppose avoir reçu $y=c+e$ et on veut retrouver c ou f.

L'idée va être de calculer un polynôme $\Lambda \in \mathbb{F}_q[X]$ tel que pour tout $i \in [\![1,n]\!]$ vérifiant $e_i \neq 0$, on a $\Lambda(e_i)=0$.
Un tel polynôme vérifie $\forall i \in [\![1,n]\!], \Lambda(x_i)y_i = \Lambda(x_i)f(x_i)$.



Maintenant on linéarise. On pose $N=\Lambda f$ et on cherche $\Lambda \in \mathbb{F}_q[X]$ de degré $\leqslant t$ et $N \in \mathbb{F}_q[X]$ de degré $\leqslant k-1+t$, et on résout

$$\forall i \in [\![1,n]\!], \Lambda(x_i)y_i = N(x_i).$$

Il s'agit d'un système linéaire à n équations et $t+1+k+t = k+2t+1$ inconnues.

Théorème 5.3

Soit $y=c+e$ où $c=(f(x_1),...,f(x_n))$ avec $f \in \mathbb{F}_q[X]$ de degré $<k$ et $w_H(e) \leqslant t = \lfloor \frac{n-k}{2} \rfloor$.
Pour tout couple $(\Lambda, N) \in \mathbb{F}_q[X]_{\leqslant t} \times \mathbb{F}_q[X]_{\leqslant k-1+t} \setminus \{0,0\}$ tels que $\forall i \in [\![1,n]\!], y_i\Lambda(x_i) = N(x_i)$ on a $\frac{N}{\Lambda} = f$.

Remarque

Le couple $(\Lambda_0, \Lambda_0 f)$ où $\Lambda_0 = \prod_{\substack{i \in [\![1,k]\!] \\ e_i \neq 0}} (X-x_i)$ vérifie la propriété.

Preuve.
Soient (Λ_1, N_1) et (Λ_2, N_2) deux solutions du système.

Fait 5.4

Si $(\Lambda_i, N_i) \neq (0,0)$ alors $\Lambda_i \neq 0$.

Preuve.
Sinon on aurait $\forall j \in [\![1,n]\!], N_i(x_j)=0$ or $\deg N \leqslant k-1+t \leqslant k-1+\frac{n-k}{2} \leqslant \frac{n}{2} - 1\frac{k}{2} \leqslant n-1$.

Fait 5.5

$$\frac{N_1}{\Lambda_1} = \frac{N_2}{\Lambda_2}$$

Preuve.
Le polynôme $N_1\Lambda_2 - \Lambda_1 N_2$ a pour racines tous les x_i, donc n racines, et a un degré $\leqslant t+k-1+t \leqslant k-1+n-k \leqslant n-1$.
Donc $N_1\Lambda_2 - \Lambda_1 N_2 = 0$.

Algorithme

Entrée : $\mathcal{C} \in \mathcal{RS}_k(x)$, $y \in \mathbb{F}_q^n$ tel qu'il existe $c \in \mathcal{RS}_k(x)$ et e de poids $\leqslant \frac{n-k}{2} = t$ tels que $y = c+e$.
Sortie : $c \in \mathcal{C}$ tel que $d(c,y) \leqslant \frac{n-k}{2}$ s'il existe, $\perp$ sinon

1. Résoudre le système.
2. Si le système n'a pas de solution $\neq (0,0)$, renvoyer $\perp$, sinon choisir une solution $(\Lambda_0, N_0) \neq (0,0)$ et renvoyer $(\frac{N_0}{\Lambda_0}(x_1), ..., \frac{N_0}{\Lambda_0}(x_n))$.

6 | Code cyclique, code BCH

Définition 6.1

Soit $\sigma : \begin{array}{ccc} \mathbb{F}_q^n & \longrightarrow & \mathbb{F}_q^n \\ (x_1, ..., x_n) & \longmapsto & (x_n, x_1, ..., x_{n-1}) \end{array}$.
Un code cyclique est un code stable par σ.

Exemple 6.2

Soit $\alpha \in \mathbb{F}_q^*$ qui engendre le groupe $\mathbb{F}_q^*$. Soit $x = (1, \alpha, .., \alpha^{q-2})$. Alors pour tout $k \leqslant q-1$, le code $\mathcal{RS}_k(x)$ est cyclique.
En effet, soit $c = (f(1), f(\alpha), ..., f(\alpha^{q-2})) \in \mathcal{RS}_k(x)$ pour $f \in \mathbb{F}_q[X]_{<k}$.
Alors avec $g = \alpha f$,

$$\begin{aligned} \sigma^{-1}(c) &= (f(\alpha), f(\alpha^2), ..., f(1)) \\ &= (g(1), g(\alpha), ..., g(\alpha^{q-2})) \end{aligned}$$

6.1 | Structure algébrique

Soit $m = (m_0, ..., m_{n-1}) \in \mathbb{F}_q^n$. On lui associe $m_0 + m_1 X + .. + m_{n-1}X^{n-1} \in \mathbb{F}_q[X]_{<n}$.
Le décalage cyclique correspond à la multiplication par X suivi de la réduction modulo $X^n - 1$.
Formellement on a l'isomorphisme

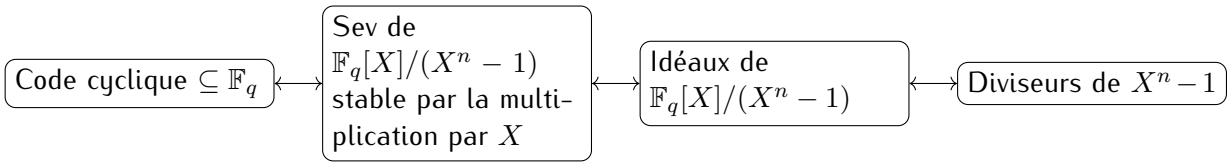
$$\begin{array}{ccc} \mathbb{F}_q^n & \xrightarrow{\varphi} & \mathbb{F}_q[X]/(X^n-1) \\ (m_0, ..., m_{n-1}) & \longmapsto & m_0 + m_1 X + ... + m_{n-1}X^{n-1} \end{array}$$

tel que le diagramme suivant commute.

$$\begin{array}{ccc} \mathbb{F}_q^n & \xrightarrow{\varphi} & \mathbb{F}_q[x]/(X^n-1) \\ \sigma \downarrow & & \downarrow \times X \\ \mathbb{F}_q^n & \xrightarrow{\varphi} & \mathbb{F}_q[x]/(X^n-1) \end{array}$$

Avec $A \in \mathbb{F}_q[X]$ premier avec $X^n - 1$, Bézout donne $U, V \in \mathbb{F}_q[X]$ tels que $AU + V(X^n - 1) =$ donc $AU \equiv 1 \mod X^n - 1$.

Voir `https://youtu.be/w6PVvDOc1dE` pour d'autres explications.



Théorème 6.3 — **Existence du polynôme générateur**

Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ un code cyclique de dimension k.
Alors il existe $g \in \mathcal{C}$ tel que $(g, \sigma(g), ..., \sigma^{k-1}(g))$ est une trace de C.
Le mot g est appelé polynôme générateur de $\mathcal{C}$.

Preuve.
Comme le code est de dimension k il existe $g \in \mathcal{C}$ non nul dont les $k-1$ dernières coordonnées sont nulles (par pivot de Gauss). De fait, $\deg(g) \leqslant n - k$. Considérons maintenant la matrice

$$\begin{pmatrix} g_0 & \cdots & g_{n-k} & 0 & \cdots & \cdots & 0 \\ 0 & g_0 & \cdots & g_{n-k} & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & & \ddots & & \vdots \\ 0 & \cdots & 0 & g_0 & \cdots & \cdots & g_{n-k} \end{pmatrix}$$

La matrice est échelonnée donc de rang k. Elle est donc matrice génératrice de $\mathcal{C}$.

Un code cyclique de dimension k a donc un unique polynôme générateur unitaire, de degré $n - k$.

6.2 | Racines d'un code cyclique, code BCH

Dans cette partie n est supposé premier avec q.

6.2.1 | Racines d'un code cyclique

Soit $\mathcal{R}_n(\mathbb{F}_q) \supseteq \mathbb{F}_q$ le corps qui contient les racines de $X^n - 1$.

Définition 6.4 — **Racines d'un code cyclique**

Soit $\mathcal{C}$ un code cyclique de générateur $g \mid X^n - 1$.
Les racines de $\mathcal{C}$ sont les racines de g dans $\mathcal{R}_n(\mathbb{F}_q)$.

Proposition 6.5

Soit $g \in \mathbb{F}_q[X]$ tel que $g \mid X^n - 1$.
Alors pour tout $\xi \in \mathcal{R}_n(\mathbb{F}_q)$ tel que $g(\xi) = 0$, on a $g(\xi^q) = 0$.

Réciproquement, soit $g \in \mathcal{R}_n(\mathbb{F}_q)[X]$ tel que $g \mid X^n - 1$ et les racines de g sont stables par $\xi \mapsto \xi^q$.
Alors $g \in \mathbb{F}_q[X]$.

Preuve.
Soit $g = \sum a_i X^i$, $a_i \in \mathbb{F}_q$.
$g(\xi) = 0 = \sum a_i \xi^i$ et $g(\xi)^q = 0 = \left(\sum a_i \xi^i\right) = \sum a_i^q \xi^{qi}$. D'autre part, comme $a_i \in \mathbb{F}_q$, $a_i^q = a_i$. D'où $\sum a_i \xi^{qi} = g(\xi^q) = 0$.

Réciproquement, soit $g = a_0 + a_1 X + ... + a_m X^m = a_m \prod_{i=1}^{m} (X - r_i)$.
Les coefficients de g sont des fonctions symétriques des racines de g : $\frac{a_0}{a_m} = (-1)^m \prod_{i=1}^m r_i$, ..., $\frac{a_{m-1}}{a_m} = -\sum_{i=1}^m r_i$.

Autrement dit, pour tout i, $a_i^q = a_i$ implique $a_i \in \mathbb{F}_q$.

Remarque 6.6

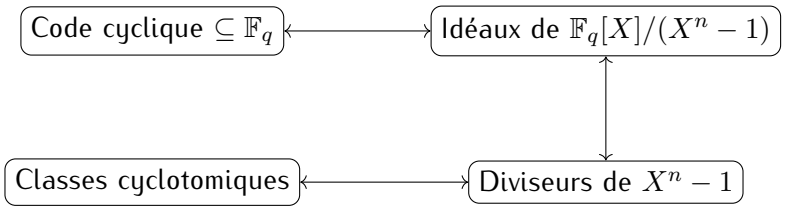
$X^n - 1 = \prod_{i=0}^{n-1}(X - \xi^i)$ où $\xi \in \mathcal{R}_n(\mathbb{F}_q)$ est une racine primitive n-ième de l'unité.
Donc les racines de $X^n - 1$ forment un groupe cyclique

$$((1, \xi, \xi^2, ..., \xi^{n-1}), \times) \simeq (\mathbb{Z}/n\mathbb{Z}, +).$$

Via cet isomorphisme, les parties de $\{1, \xi, ..., \xi^{n-1}\}$ stables par $x \mapsto x^q$ s'identifient aux parties de $\mathbb{Z}/n\mathbb{Z}$ stables par $u \mapsto qu$.

Définition 6.7 **Classe cyclotomique**

Une classe cyclotomique est un sous-ensemble de $\mathbb{Z}/n\mathbb{Z}$ stable par multiplication par q.



Exemple 6.8 **Cas particulier**

Si $n = q - 1$, alors $\mathcal{R}_n(\mathbb{F}_q) = \mathbb{F}_q$.
En effet, $X^{q-1} - 1 = \prod_{\alpha \in \mathbb{F}_q^*}(X - \alpha)$ car $X^q - X = \prod_{\alpha \in \mathbb{F}_q}(X - \alpha)$.
Dans ce cas toute partie de $\mathbb{Z}/n\mathbb{Z}$ est une classe cyclotomique. En effet dans $\mathbb{Z}/(q-1)\mathbb{Z}$, $x \mapsto qx$ est l'identité.

6.2.2 | Codes BCH

Théorème 6.9 **Borne BCH**

Soit $\mathcal{C} \subseteq \mathbb{F}_q^n$ le code cyclique associé à la classe cyclotomique $I \subseteq \mathbb{Z}/n\mathbb{Z}$.
Si I contient une suite de la forme $a, a+1, ..., a+s-1$, alors $d_{\min}(\mathcal{C}) \geqslant s+1$.

Preuve.
Soit $(c_0, ..., c_{n-1}) \in \mathcal{C} \setminus \{0\}$ et $c(X) \in \mathbb{F}_q[X]/(X^n - 1)$ le polynôme correspondant.
On a $c(\xi^a) = c(\xi^{a+1}) = ... = c(\xi^{a+s-1}) = 0$.
Supposons d'aventure que $w_H(c) = r \leqslant s$. Soient $i_0, ..., i_{n-1}$ les indices tels que $c_{i_j} \neq 0$.
De fait,

$$\begin{cases} c_{i_0}\xi^{ai_0} + c_{i_1}\xi^{ai_1} + ... + c_{i_{n-1}}(\xi^{ai_{n-1}}) = 0 \\ \vdots \\ c_{i_0}\xi^{(a+s-1)i_0} + ... + c_{i_{n-1}}(\xi^{(a+s-1)i_{n-1}}) = 0 \end{cases}$$

Donc

$$\underbrace{\begin{pmatrix} 1 & \cdots & 1 \\ \xi^{i_0} & \cdots & \xi^{i_{n-1}} \\ \xi^{2i_0} & \cdots & \xi^{2i_{n-1}} \\ \vdots & \ddots & \vdots \\ \xi^{(s-1)i_0} & \cdots & \xi^{(s-1)i_{n-1}} \end{pmatrix}}_{M} \cdot \begin{pmatrix} c_{i_0}\xi^{ai_0} \\ \vdots \\ c_{i_{n-1}}\xi^{ai_{n-1}} \end{pmatrix} = 0$$

Donc $\begin{pmatrix} c_{i_0}\xi^{ai_0} \\ \vdots \\ c_{i_{n-1}}\xi^{ai_{n-1}} \end{pmatrix}$ est non nul par hypothèse et appartient à $\ker M$.
Or, $\mathrm{Rg}\, M = r$ car le mineur max associé aux r premières lignes est un déterminant de Vandermonde non nul.
Donc on a $\ker M = \{0\}$. Fichtre ! Diantre ! Vertuchou ! L'aventure est finie.

Définition 6.10 **Code BCH**

Soit $\delta > 0$.
On définit le code $\mathrm{BCH}_{q,n}(a, \delta)$ comme le code cyclique associé à la plus petite classe cyclotomique contenant $a, a+1, ..., a+\delta-2$.

Proposition 6.11

$\mathrm{BCH}_{q,n}(a, \delta)$ a une distance minimale $\geqslant \delta$.

Exemple 6.12

Pour $q = 2$ et $n = 15$, les classes cyclotomiques minimales sont $\{0\}$, $\{1, 2, 4, 8\}$, $\{3, 6, 12, 9\}$, $\{5, 10\}$ et $\{7, 14, 13, 11\}$.
Par le calcul,

$$\begin{aligned} \{0\} &\longleftrightarrow X - 1 = g_0 \\ \{1, 2, 4, 8\} &\longleftrightarrow 1 + X + X^4 = g_1 \\ \{3, 6, 12, 9\} &\longleftrightarrow 1 + X + X^2 + X^3 + X^4 = g_3 \\ \{5, 10\} &\longleftrightarrow 1 + X + X^2 = g_5 \\ \{7, 14, 13, 11\} &\longleftrightarrow (X^{15} - 1)/\text{les autres} = g_7 \end{aligned}$$

δ	Classe	k	g
3	$\{1, 2, 4, 8\}$	11	g_1
4	$\{0, 1, 2, 4, 8\}$	10	$g_0 g_1$
5	$\{1, 2, 3, 4, 6, 8, 9, 12\}$	7	$g_1 g_3$

Exemple 6.13 **Retour au cas particulier**

Dans le cas $n = q - 1$, toute partie de $\mathbb{Z}/n\mathbb{Z}$ est une classe cyclotomique.
Considérons $\mathrm{BCH}_{q,n}(0, \delta)$, i.e. le code cyclique associé à la classe $\{0, ..., \delta - 2\}$.
Par la borne BCH il a pour paramètres $[n, n-\delta+1, \geqslant \delta]$, et par la borne de Singleton on a $d_{\min} = \delta$.
En fait on retrouve les codes de Reed-Solomon.

Deuxième partie

Anne Canteaut

7 | Fonctions booléennes et codes de Reed-Muller

Définition 7.1 **Fonction booléenne**

Une fonction booléenne à n variables est une fonction de $\mathbb{F}_2^n$ dans $\mathbb{F}_2$.
Son vecteur de valeurs est le vecteur binaire de longueur 2^n $(f(x) \mid x \in \mathbb{F}_2^n)$.

Exemple 7.2

Pour $n = 3$ on aura tous les triplets :

(x_1, x_2, x_3)	000	100	010	110	001	101	011	111
$f(x_1, x_2, x_3)$	0	1	0	0	0	1	1	1

Définition 7.3 Poids des valeurs

Avec v_f le vecteur binaire de f, on appelle poids de f son poids de Hamming : $wt(f) = wt(v_f)$.
f est dite équilibrée si $wt(f) = 2^{n-1}$.

7.1 Forme algébrique normale (ANF)

On considère l'espace $\mathbb{F}_2[x_1, ..., x_n]/(x_1^2 + x_1, ..., x_n^2 + x_n)$. On a alors $x_i^2 = x_i$, alors on ne peut avoir que des termes de degré 0 ou 1.

Notation 7.4 Monômes

Avec $u \in \mathbb{F}_2^n$ on note $x^u \prod_{i=1}^{n} x_i^{u_i}$.

Par exemple pour $n = 4$, $x^{1011} = x_1 x_3 x_4$.

Proposition 7.5

Toute fonction booléenne à n variables s'écrit de manière unique sous forme algébrique normale

$$\sum_{u \in \mathbb{F}_2^n} a_u x^u, \quad a_u \in \mathbb{F}_2.$$

Preuve.
La preuve se fait par récurrence sur n.

Pour $n = 1$ on a $f(x) = f(0) + (f(0) + f(1))x$.

Par récurrence, on peut construire $f(x_1, ..., x_{n-1}, 0) = \sum_{u \in \mathbb{F}_2^{n-1}} a_u x^u = g(x_1, ..., x_{n-1})$ et $f(x_1, ..., x_{n-1}, 1) = \sum_{u \in \mathbb{F}_2^{n-1}} b_u x^u = h(x_1, ..., x_{n-1})$.

On peut alors écrire $f(x_1, ..., x_n) = (1 + x_n)g(x_1, ..., x_{n-1}) + x_n h(x_1, ..., x_{n-1})$.

Proposition 7.6 Transformation de Möbius

Soit f une fonction booléenne à n variables. On note son ANF $\sum_{u \in \mathbb{F}_2^n} a_u x^u$.
Alors $\forall x \in \mathbb{F}_2^n, f(x) = \sum_{u \preccurlyeq x} a_u$, où $u \preccurlyeq x$ ssi $\forall i, u_i \leqslant x_i$.

Par exemple $f(101) = a_{101} + a_{100} + a_{001} + a_{000}$.

Exemple 7.7

En reprenant l' ?? 7.2, on le découpe en paquets de deux :
$0\ \ 1 \mid 0\ \ 0 \mid 0\ \ 1 \mid 1\ \ 1$
et on remplace chaque dernier bit par la somme des bits du paquet :
$0\ \ 1 \mid 0\ \ 0 \mid 0\ \ 1 \mid 1\ \ 0$
on découpe en paquets de quatre et on fait la même chose :
$0\ \ 1\ \ 0\ \ 1 \mid 0\ \ 1\ \ 1\ \ 1$

et enfin avec le bloc complet :
$0\quad 1\quad 0\quad 1\quad 0\quad 1\quad 1\quad 0 = (a_u, u \in \mathbb{F}_2^n)$.
Alors $f(x_1, x_2, x_3) = x_1 + x_1x_2 + x_2x_3$.

7.2 | Codes de Reed-Muller (1954)

Définition 7.8 Codes de Reed-Muller

Soit $m > 0$ et $0 \leqslant r \leqslant m$ deux entiers.
Le code de RM de longueur 2^m et d'ordre r, noté $\mathcal{R}(r, m)$, est l'ensemble des vecteurs des valeurs des fonctions booléennes à m variables de degré $\leqslant r$:

$$\mathcal{R}(r, m) = \{(f(x), x \in \mathbb{F}_2^m) \mid f : \mathbb{F}_2^m \to \mathbb{F}_2, \deg f \leqslant r\}.$$

Ici contrairement aux codes de Reed-Solomon, les polynômes sont multivariés et à valeurs dans un petit corps.

Exemple 7.9

La matrice génératrice du code $\mathcal{R}(1, 3)$ est la suivante :

$$G = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 000 & 100 & 010 & 110 & 001 & 101 & 011 & 111 \end{bmatrix} \begin{matrix} 1 \\ x_1 \\ x_2 \\ x_3 \end{matrix}$$

Remarque 7.10

— $\dim \mathcal{R}(r, m) = \sum_{i=0}^{r} \binom{m}{i}$
— $\mathcal{R}(r-1, m) \subseteq \mathcal{R}(r, m)$

Proposition 7.11

$\mathcal{R}(r, m) = \{u \mid u + v) \mid u \in \mathcal{R}(r, m-1), v \in \mathcal{R}(r-1, m-1)\}$

$$G(r, m) = \left[\begin{array}{c|c} G(r, m-1) & G(r, m-1) \\ \hline 0 & G(r-1, m-1) \end{array}\right]$$

Preuve de la ?? 7.11.
$f(x_1, ..., x_m) = \underbrace{g(x_1, ..., x_{m-1})}_{\deg g \leqslant r} + x_m \underbrace{h(x_1, ..., x_{m-1})}_{\deg h \leqslant r}$

$$v_f = \begin{array}{c} \\ (\\ (\end{array} \begin{array}{c|c} x_m = 0 & x_m = 1 \\ & \\ v_g & v_g + v_h \end{array} \begin{array}{c} \\ ) \\ ) \end{array}$$

Proposition 7.12

Soient $\mathcal{C}_1$ et $\mathcal{C}_2$ deux codes binaires (m, M_1, d_1) et (m, M_2, d_2).
Alors avec $\mathcal{C} = \{(u \mid u + v) \mid u \in \mathcal{C}_1, v \in \mathcal{C}_2\}$, on a un code $(2m, M_1M_2, d)$ où $d = \min(2d_1, d_2)$.

Preuve.
Avec $c = (u \mid u + v)$, $c' = (u' \mid u' + v')$, on a

— si $v = v'$, $d(c, c') = 2d(u, u') \geqslant 2d_1$
— si $v \neq v'$, $d(c, c') = wt(u + u') + wt(u + u' + v + v') \geqslant wt(v + v') \geqslant d_2$
car

$$wt((u + u') + (v + v')) = wt(u + u') + wt(v + v') - \underbrace{2wt((u + u') \cap (v + v'))}_{\leqslant wt(u+u')}$$
$$\geqslant wt(v + v') - wt(u + u').$$

Proposition 7.13

$d_{\min}(\mathcal{R}(r, m)) = 2^{m-r}$

Preuve.
La preuve est par récurrence sur m.

Pour $m = 1$ on a pour $r = 0$, $\mathcal{R}(0, 1) = \{(00), (11)\}$ donc $d_{\min} = 2$, et pour $r = 1$, $\mathcal{R}(1, 1) = \{(00), (10), (01), (11)\} = \mathbb{F}_2^2$ donc $d_{\min} = 1$.

Par récurrence, on a

$$\begin{aligned} d_{\min}(\mathcal{R}(r, m)) &= \min(2d_{\min}\mathcal{R}(r, m-1), d_{\min}(\mathcal{R}(r-1, m-1)) \\ &= \min(2 \times 2^{m-1-r}, 2^{m-r}) \\ &= 2^{m-r}. \end{aligned}$$

7.3 | Distribution des poids de $\mathcal{R}(r, m)$

Les cas triviaux sont
— $\mathcal{R}(0, m) = \{(0...0), (1...1)\}$
— $\mathcal{R}(m, m) = \mathbb{F}_2^m$.

Proposition 7.14

$\mathcal{R}(1, m)$ est composé de $(0...0)$ et $(1...1)$ et de $(2^{m+1} - 2)$ mots équilibrés (i.e. de poids 2^{m-1}).
$\dim \mathcal{R}(1, m) = \sum_{i=0}^{1} \binom{m}{i} = m + 1.$

Preuve.
Les mots non constants de $\mathcal{R}(1, m)$ sont les vecteurs des valeurs des fonctions de degré 1, $f(\underbrace{x_1, ..., x_n}_{x}) = \underbrace{a \cdot x}_{\sum a_i x_i} + \underbrace{\varepsilon}_{\in \mathbb{F}_2}$.
$wt(f) = \#\{x \in \mathbb{F}_2^m, a \cdot x = \varepsilon + 1\}$ donc si $\varepsilon = 1$ on a $wt(f) = \#\langle a \rangle^\perp = 2^{m-1}$ et si $\varepsilon = 0$ alors $wt(f) = \#\mathbb{F}_2^m \setminus \langle a \rangle^\perp = 2^{m-1}$.

L'autre code intéressant est $\mathcal{R}(m - 1, m)$ qui est un code $[2^m, 2^m - 1, 2]$.

Proposition 7.15

$\mathcal{R}(m - 1, m)$ est l'ensemble des mots de longueur 2^m et de poids pair.

Preuve.
On peut le montrer par récurrence sur m, on utilise que $wt(a + b) = wt(a) + wt(b) - 2wt(a \cap b)$.
Sinon on peut utiliser le fait que le poids des monômes est pair.

On utilise en crypto le fait que f à n variables est de poids impair ssi $\deg f = n$, pour justifier de ne pas choisir des fonctions de degré maximal.

On remarque que $\mathcal{R}(m - 1, m) = \langle (1...1) \rangle^\perp = \mathcal{R}(0, m)^\perp$. Ce résultat est plus général.

Proposition 7.16

$\mathcal{R}(r,m)^\perp = \mathcal{R}(m-r-1,m)$

Preuve.
Avec $c = v_f \in \mathcal{R}(r,m)$ et $c' = v_g \in \mathcal{R}(m-r-1,m)$,

$$\begin{aligned} c \cdot c' &= \sum_{i=1}^{m} c_i c'_i \mod 2 \\ &= \underbrace{wt(c_i c'_i)}_{\substack{\text{vecteur} \\ \text{des valeurs} \\ \text{de } f \times g}} \mod 2 \\ &= 0 \end{aligned}$$

car $\deg f \times g \leqslant r + m - r - 1 = m - 1$.

8 | Attaques sur les chiffrements par bloc

Pour opérer sur des données de taille arbitraire, on aime souvent traiter des blocs de taille fixe et répéter l'opération.

Notation 8.1

On note la fonction de chiffrement comme suit :

$$E : \begin{array}{ccc} \mathbb{F}_2^n \times \mathbb{F}_2^\kappa & \longrightarrow & \mathbb{F}_2^n \\ (m,k) & \longmapsto & c = E(n,k) = E_k(m) \end{array}$$

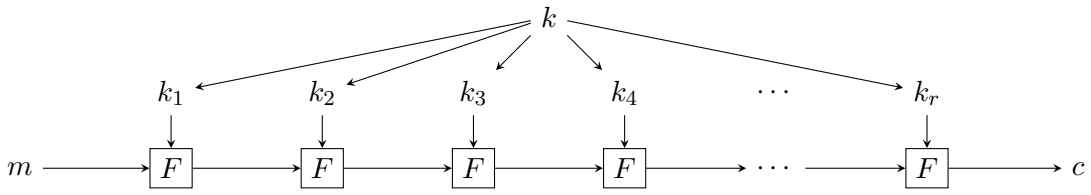
n correspond à la taille des blocs. Usuellement on a $n \in \{64, 128, 256\}$.
κ correspond à la taille de la clé. Usuellement on a $\kappa \in \{128, 256\}$.

Définition 8.2 Mode CTR

Soit $m = m_1 \cdot \ldots \cdot m_N$ un message, avec m_i des blocs de n bits.
Pour chiffrer m on tire un nonce N, et le chiffrement du bloc m_i est $m_i \oplus E(i \| \mathrm{N}, k)$.

Le mode CTR est très uilisé en pratique. La sécurité est assurée si les données sont beaucoup plus petites que $2^{n/2}$ blocs. Avec des blocs de 64 bits on a donc une limite de quelques Go, ce qui est aujourd'hui parfois limitant.

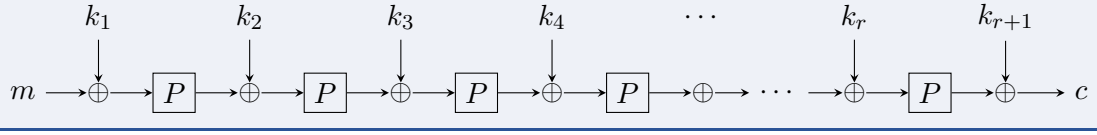
On veut que $\{E_k \mid k \in \mathbb{F}_2^\kappa\}$ soit indistingable des 2^κ permutations de $\mathbb{F}_2^n$ choisies aléatoirement.
Pour essayer d'atteindre cela, on va itérer une fonction simple F.



On peut également utiliser le chiffrement suivant.

Définition 8.3 Key-alternating cipher

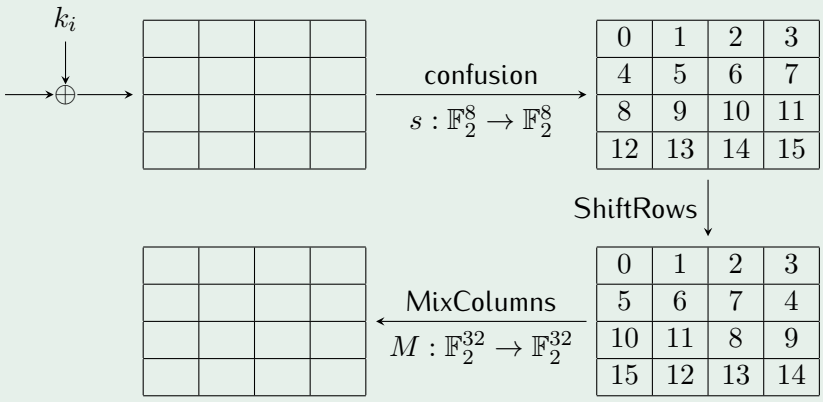
Avec P une permutation, le key-alternating cipher est le suivant :



Pour éviter de retrouver facilement la permutation utilisée, Shannon a décrit deux propriétés : la confusion et la diffusion. La confusion mesure la non-linéarité, mais il est compliqué d'implémenter des fonctions non linéaires. Avec $n = m \times p$ et $m \in \{4, 5, 8\}$, on a une fonction de confusion non-linéaire $s : \mathbb{F}_2^m \to \mathbb{F}_2^m$.

Exemple 8.4 AES-128

Ici la clé a 128 bits et $r = 10$.



8.1 Attaques algébriques

On considère $E : \begin{array}{ccc} \mathbb{F}_2^{n+\kappa} & \longrightarrow & \mathbb{F}_2^n \\ (m_1, ..., m_n, k_1, ..., k_\kappa) & \longmapsto & (f_1, ..., f_n) \end{array}$, avec f_i des fonctions booléennes à n variables (clair) et κ variables (clef).
On a alors un système de n équations booléennes à κ inconnues.

$$\begin{cases} c_1 = f_1(m_1, ..., m_n, k_1, ..., k_\kappa) \\ c_2 = f_2(m_1, ..., m_n, k_1, ..., k_\kappa) \\ \dots \\ c_n = f_n(m_1, ..., m_n, k_1, ..., k_\kappa) \end{cases}$$

Il faut donc E de degré élevé. Le critère qui en suit est qu'il faut s de degré le plus élevé possible.

$$s : \begin{array}{ccc} \mathbb{F}_2^8 & \longrightarrow & \mathbb{F}_2^8 \\ (x_1, ..., x_8) & \longmapsto & \begin{array}{c} s_1(x_1, ..., x_8) \\ s_2(x_1, ..., x_8) \\ \dots \\ s_8(x_1, ..., x_8) \end{array} \end{array}$$

Si les s_i sont équilibrés on a $wt(s_i) = 2^7$ donc les s_i sont de degré $\leqslant 7$.

Dans l'AES, les représentations possibles de s sont

— collection de 8 fonctions booléennes à 8 variables

— $s : \mathbb{F}_2^8 \to \mathbb{F}_2^8$, or $\mathbb{F}_2^8 \sim \mathbb{F}_{2^8}$, et $S : \mathbb{F}_{2^8} \to \mathbb{F}_{2^8}$, alors il existe un unique polynôme de $\mathbb{F}_{2^8}[X]/(X^{2^8} + X)$ $\sum_{i=0}^{2^8-1} A_i X^i$

— $\begin{array}{ccc} \mathbb{F}_2^8 & \longrightarrow & \mathbb{F}_{2^8} \\ (x_0, ..., x_7) & \longmapsto & \sum_{i=0}^7 x_i \alpha^i \end{array}$, où α est racine de $X^8 + X^4 + X^3 + X + 1$

— $S : \begin{array}{ccc} \mathbb{F}_{2^8} & \longrightarrow & \mathbb{F}_{2^8} \\ x & \longmapsto & x^{254} \end{array}$, qui est la fonction inverse de $\mathbb{F}_{2^8}$

— $\mathbb{F}_{2^8} \to \mathbb{F}_2^8$ une fonction affine sur $\mathbb{F}_2^8$.

Proposition 8.5

$$S(X) = \sum_{i=0}^{2^n-1} A_i X^i$$
$$\deg S = \max\{wt(i) \mid A_i \neq 0\}$$
$$S(X) = X^{254}$$

Attaques exploitant des relations des entrées et sorties de F

$$S : \begin{array}{ccc} \mathbb{F}_2^m & \longrightarrow & \mathbb{F}_2^m \\ (x_1, ..., x_m) & \longmapsto & (s_1(x), ..., s_m(x)) \end{array}$$

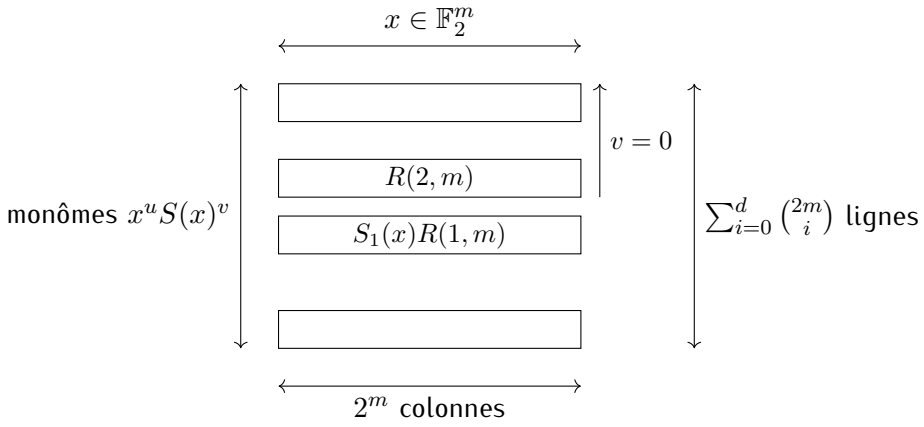
Exemple 8.6

Pour $m = 4$, $S : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ est de degré 3.
$\forall x \in \mathbb{F}_2^4, x_2x_4 + x_2s_1(x) + x_2s_2(x) = 0$

On cherche

$$\forall x, \sum_{\substack{u,v \in \mathbb{F}_2^m \\ w(u)+w(v) \leqslant d}} c_{u,v} x^u [S(x)]^v = 0.$$

Et on veut savoir pour quelle valeur de d on peut trouver ça. Une façon de faire c'est de regarder le vecteur des valeurs des monômes $(x^u S(x)^v)$.



Proposition 8.7

Pour toute $S : \mathbb{F}_2^m \to \mathbb{F}_2^m$, le nombre de relations indépendantes de degré $\leqslant d$ est au moins $\sum_{i=0}^{d} \binom{2m}{i} - 2^m$.

Exemple 8.8

Pour $m = 8$, avec $d = 3$ on a au moins 441 relations de degré 3.
Donc on peut toujours avoir des relations de degré 3, et non 7.

Dans le cadre de l'AES on a que S est la fonction inverse, donc $\forall x \neq 0, xS(x) = 1$, donc

$$\forall x \in \mathbb{F}_{2^m}, x^2 S(x) = x.$$

Et sur $\mathbb{F}_2^m$ on obtient 8 relations booléennes entre $x \in \mathbb{F}_2^8$ et $S(x)$ de degré 2 car le carré est une opération de degré 1 dans un corps de caractéristique 2.
Avec $m = 8$, pour l'AES, on a 39 relations indépendantes de degré 2.

Pour retrouver la clef à partir d'un système polynômial de degré 2, on va récupérer un système de degré 2 à 1280 inconnues et 8000 équations, ce qui est trop gros pour être résolu.

8.2 Attaques statistiques

Définition 8.9 Distingueur

Un distingueur est une fonction qui a une propriété satisfaite par E_k, pour $k \in \mathbb{F}_2^\kappa$, qui est satisfaite avec probabilité p sur les entrées de E_k, qui diffère significativement de la probabilité que la propriété soit satisfaite pour une permutation aléatoire.

On a E_k : $\begin{array}{ccc} \mathbb{F}_2^n & \longrightarrow & \mathbb{F}_2^n \\ (x_1,...,x_n) & \longmapsto & (y_1,...,y_n) \end{array}$.
Pour α fixé, on a un distingueur si $\mathbb{P}_{x\in\mathbb{F}_2^n}(y_1 = \alpha \cdot x) \neq \frac{1}{2}$.

Le distingueur va prendre des couples clair-chiffré et renvoyer 0 s'il s'agit d'une permutation de $\mathbb{F}_2^n$, et 1 si c'est une fonction E_k.

$$(x_i, \pi(x_i))_{1\leqslant i\leqslant N} \longrightarrow \boxed{\mathcal{D}} \longrightarrow \begin{array}{l} 0 \text{ si } \pi \text{ permutation de } \mathbb{F}_2^n \\ 1 \text{ si } \pi \in \{E_k \mid k \in \mathbb{F}_2^\kappa \end{array}$$

Définition 8.10 Complexité en données

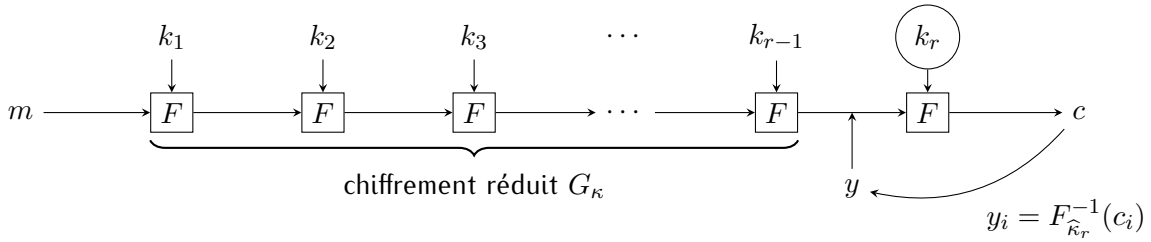
On définit l'avantage du distingueur par

$$\mathrm{Adv}(\mathcal{D}) = \mathbb{P}(\mathcal{D}=1 \mid \pi \in \{E_k \mid k \in \mathbb{F}_2^\kappa\}) - \mathbb{P}(\mathcal{D}=0 \mid \pi \in \{E_k \mid k \in \mathbb{F}_2^\kappa\}).$$

Quand on a un chiffrement par bloc, on ne veut pas qu'il y ait un distingueur facile à évaluer.

8.2.1 Attaques sur le dernier tour

On va essayer de retrouver la clé du dernier tour.



Pour chaque valeur $\widehat{\kappa}_r$
  Pour chaque (m_i, c_i)
    $y_i \leftarrow F_{\widehat{\kappa}_r}^{-1}(c_i)$
  Appliquer $\mathcal{D}$ aux $(m_i, y_i)_{1\leqslant i\leqslant N}$
  Si $\mathcal{D} = 1$, retrouver $\widehat{\kappa}_r$

(c'est pertinent si $|\kappa_r| < |\kappa|$ ou si $\mathcal{D}$ ne nécessite pas tous les bits de y)

8.2.2 Cryptanalyse linéaire

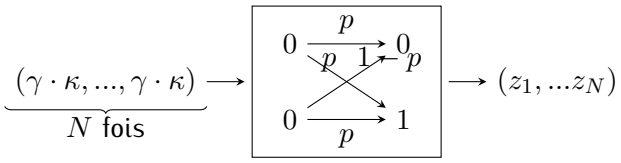
Le distingueur est une relation de degré 1 sur $\mathbb{F}_2$ entre les entrées, sorties et clef de G_κ.
$G_\kappa : (x_1,...,x_n) \longmapsto (y_1,...,y_n) = G_\kappa(x)$

$\mathbb{P}_x(\alpha \cdot x + \beta \cdot G_\kappa(x) + \gamma \cdot \kappa = 0) = \dfrac{1}{2}(1+\varepsilon)$, avec $\alpha, \beta \in \mathbb{F}_2^n$. α est le masque d'entrée, β est le masque de sortie, et $\varepsilon > 0$ est le biais.

Complexité du distingueur

Avec la probablité $p = \mathbb{P}_x(\alpha \cdot x + \beta \cdot G_\kappa(x) + \gamma \cdot \kappa = 0) > \frac{1}{2}$, on veut retrouver $\gamma \cdot \kappa$.
De combien de couples $x_i, E_k(x_i)$ a-t-on besoin ?

On utilise un code de répétition.

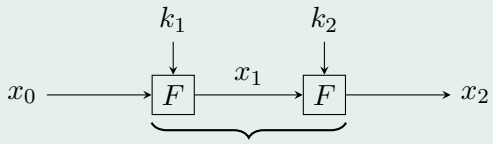


On a $\mathbb{P}_i(z_i = \gamma \cdot \kappa) = p$.

Par théorème de Shannon on a que le taux de transmission, $\frac{1}{N}$ est inférieur à $C_{\text{BSC}}(1-p) \simeq \frac{\varepsilon^2}{\ln 2}$. Donc pour retrouver $\gamma \cdot \kappa$ il faut $N \geqslant \frac{\ln 2}{\varepsilon^2}$ couples clair-chiffré.
Pour attaquer sur le dernier tour, il faut $\frac{\ln 2 \times \text{nombres de bits de } \kappa_r \text{ impliqués}}{\varepsilon^2}$. Donc c'est inutilisable.

Chaîner les approximations linéaires

Exemple 8.11 — **Deux itérations de F**



$$\alpha \cdot x_0 + \beta \cdot x_1 + \gamma_1 \cdot k_1 = 0, \quad p_1 = \tfrac{1}{2}(1+\varepsilon_1)$$
$$\beta \cdot x_1 + \delta \cdot x_2 + \gamma_2 \cdot k_2 = 0, \quad p_2 = \tfrac{1}{2}(1+\varepsilon_2)$$

Donc $\alpha \cdot x_0 + \delta \cdot x_2 + (\gamma_1 \cdot k_1 + \gamma_2 \cdot k_2) = 0$ et

$$\begin{aligned} p &= \frac{1}{4}(1+\varepsilon_1)(1+\varepsilon_2) + \frac{1}{4}(1-\varepsilon_1)(1-\varepsilon_2) \\ &= \frac{1}{4}(2+2\varepsilon_1\varepsilon_2) \\ &= \frac{1}{2}(1+\varepsilon_1\varepsilon_2). \end{aligned}$$

Proposition 8.12 — **Piling-up lemma (Matsui 94)**

Soient $X_1, ..., X_m$ des variables aléatoires *indépendantes* de Bernoulli, qui prennent la valeur 0 avec probabilité $\frac{1}{2}(1+\varepsilon_i)$ et 1 avec probabilité $\frac{1}{2}(1-\varepsilon_i)$.
Alors $\mathbb{P}[X_1 + ... + X_m = 0] = \frac{1}{2}\left(1 + \prod_{i=1}^{m} \varepsilon_i\right)$.

L'idée est de chercher des chemins linéaires (masques) $(\alpha_0, ..., \alpha_r)$ tels que l'approximation (α_i, α_{i+1}) ait un biais $|\varepsilon_i|$ élevé. On maximise $\prod_{i=1}^{m} \varepsilon_i$.

Approximations linéaires sur un tour

Soit $F : \mathbb{F}_2^n \to \mathbb{F}_2^n$. On va chercher les biais des fonctions $\begin{array}{ccc} \mathbb{F}_2^n & \longrightarrow & \mathbb{F}_2 \\ x & \mapsto & b \cdot F(x) + a \cdot x \end{array}$ pour tous a, b avec $b \neq 0$.

Définition 8.13 — **Biais d'une fonction booléenne à une variable**

Soit $f : \mathbb{F}_2^n \to \mathbb{F}_2$.

$$\begin{aligned} \mathcal{E}(f) &= \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)} \\ &= (-1)wt(f) + 2^m - wt(f) \\ &= 2^m - 2wt(f) \end{aligned}$$

On a donc $\mathbb{P}_x[f(x) = 0] = \frac{2^n - wt(f)}{2^n} = \frac{1}{2}\left(1 + \frac{\mathcal{E}(f)}{2^n}\right)$.

Définition 8.14 Transformée de Walsh

Soit $f : \mathbb{F}_2^n \to \mathbb{F}_2$. La transformée de Walsh de f est la fonction

$$\begin{array}{rcl} \mathbb{F}_2^n & \longrightarrow & \mathbb{Z} \\ a & \longmapsto & \mathcal{E}(f + \underbrace{\varphi_a}_{x \mapsto a\cdot x}) = \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)+a\cdot x} \end{array}$$

La transformée de Walsh est un peu une transformée de Fourier discrète.

On va calculer les transformations de Walsh de toutes les fonctions $x \mapsto b \cdot F(x)$ pour $b \in \mathbb{F}_2^n \setminus \{0\}$.

Propriété 8.15

La transformée de Walsh est involutive (à une constante près) :

$$\forall b \in \mathbb{F}_2^n, \sum_{a \in \mathbb{F}_2^n} (-1)^{a\cdot b} \mathcal{E}(f + \varphi_a) = 2^n (-1)^{f(b)} \cdot .$$

Preuve.
Soit $b \in \mathbb{F}_2^n$.

$$\begin{aligned} \sum_{a \in \mathbb{F}_2^n} (-1)^{a\cdot b} \mathcal{E}(f + \varphi_a) &= \sum_{a \in \mathbb{F}_2^n} \sum_{x \in \mathbb{F}_2^n} (-1)^{a\cdot b} (-1)^{f(x)+a\cdot x} \\ &= \sum_{x \in \mathbb{F}_2^n} (-1)^{f(x)} \underbrace{\left(\sum_{a \in \mathbb{F}_2^n} (-1)^{a\cdot(b+x)} \right)}_{\mathcal{E}(\varphi_{b+x}) = \begin{cases} 0 & \text{si } b+x \neq 0 \\ 2^n & \text{si } b+x = 0 \end{cases}} \\ &= (-1)^{f(b)} 2^n. \end{aligned}$$

Calcul avec FFT

Pour calculer $\mathcal{E}(f + \varphi_a)$, avec $f : \mathbb{F}_2^n \to \mathbb{F}_2$ et $a \in \mathbb{F}_2^n$, il faut naïvement du $O(2^{2n})$, mais avec une FFT il suffit de $O(n2^n)$.

Exemple 8.16

En reprenant l' ?? 7.2, avec $x = x_1 + x_1x_2 + x_2x_3$, en regroupant successivement par paquets de deux comme dans l' ?? 7.7, on a

x	000	100	010	110	001	101	011	111
$f(x)$	0	1	0	0	0	1	1	1
$(-1)^{f(x)}$	1	−1	1	1	1	−1	−1	−1
étape 1	[0	2]	[2	0]	[0	2]	[−2	0]
étape 2	[2	2	−2	2]	[−2	2	2	2]
étape 3	0	4	0	4	4	0	−4	0

À l'étape 3 on obtient tous les $\mathcal{E}(f + \varphi_a)$. Donc on a que $\max_{a \in \mathbb{F}_2^3} |\mathcal{E}(f + \varphi_a)| = 4$.
Par exemple pour $a = 100$, $\mathbb{P}_{x \in \mathbb{F}_2^3}(f(x) + x_1 = 0) = \frac{1}{2}(1 + \frac{4}{2^3}) = \frac{3}{4}$.

Proposition 8.17 Relation de Parceval

Soit $f : \mathbb{F}_2^n \to \mathbb{F}_2$.
Alors $\sum_{a \in \mathbb{F}_2^n} (\mathcal{E}(f + \varphi_a))^2 = 2^{2n}$.

Preuve.

$$\begin{aligned}\sum_{a\in\mathbb{F}_2^n}(\mathcal{E}(f+\varphi_a))^2 &= \sum_{a\in\mathbb{F}_2^n}\left(\sum_{x\in\mathbb{F}_2^n}(-1)^{f(x)+a\cdot x}\right)\left(\sum_{y\in\mathbb{F}_2^n}(-1)^{f(y)+a\cdot y}\right)\\ &= \sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^n}(-1)^{f(x)+f(y)}\underbrace{\left(\sum_{a\in\mathbb{F}_2^n}(-1)^{a\cdot(x+y)}\right)}_{\mathcal{E}(\varphi_{x+y})=0 \text{ si } x+y\neq 0}\\ &= \sum_{x\in\mathbb{F}_2^n}\sum_{y\in\mathbb{F}_2^n}(-1)^{f(x)+f(y)}2^n\\ &= 2^n 2^n.\end{aligned}$$

Définition 8.18 Linéarité d'une fonction

La linéarité de f, notée $\mathcal{L}(f)$ est définie par $\mathcal{L}(f)=\max_{a\in\mathbb{F}_2^n}|\mathcal{E}(f+\varphi_a)|$.

Proposition 8.19

Soit $f:\mathbb{F}_2^n\to\mathbb{F}_2$, alors $\mathcal{L}(f)\geqslant 2^{n/2}$.
Les fonctions qui atteignent cette borne sont appelées fonctions courbes (bent), ce qui peut arriver uniquement si n est pair.

Preuve.
Si d'aventure $\mathcal{L}(f)<2^{n/2}$, alors pour tout a, $\mathcal{E}^2(f+\varphi_a)<2^n$. Donc $\sum_{a\in\mathbb{F}_2^n}\mathcal{E}^2(f+\varphi_a)<2^{2n}$. Diantre !

L'aventure est finie.

Si n est pair on peut toujours trouver des fonctions courbes. Si n est impair on peut obtenir des fonctions telles que $\mathcal{L}(f)\leqslant 2^{(n+1)/2}$.

n	4	5	6	7	8	9	10	11
$\min\mathcal{L}(f)$	4	8	8	16	16	$\in[\![24,30]\!]$	32	$\in[\![46,60]\!]$

On peut se demander aussi la plus petite valeur de linéarité qu'on peut obtenir quand f est équilibrée.

n	4	5	6	7	8	9	10
$\min_{f\text{ équilibrée}}\mathcal{L}(f)$	8	8	12	16	$\in\{20,24\}$	$\in\{24,28,32\}$	$\in\{36,40\}$

On définit le code $\mathcal{C}_f$ comme le code de Reed-Muller $R(1,m)$ avec le vecteur des mots de f ajouté à la matrice génératrice.
Donc $\mathcal{C}_f=R(1,m)\cup(f+R(1,m))=\{\varphi_a\},\{\varphi_a+1\},\{f+\varphi_a\},\{f+\varphi_a+1\}$.
$d_{\min}(\mathcal{C}_f)=\min_{a,\varepsilon\in\mathbb{F}_2} wt(f+\varphi_a+\varepsilon)=d_H(f,R(1,m))=2^{m-1}-\frac{1}{2}\mathcal{L}(f)$.

Définition 8.20 Rayon de recouvrement

Le rayon de recouvrement d'un code $\mathcal{C}$, noté $\rho(\mathcal{C})$ est $\max_{x\in\mathbb{F}_2^n} d_H(x,\mathcal{C})$.

Définition 8.21 Linéarité d'une fonction

Soit $F:\mathbb{F}_2^n\to\mathbb{F}_2^m$. Sa linéarité $\mathcal{L}(F)$ est $\max_{b\neq 0}\mathcal{L}(F_b)$ où F_b est la fonction booléenne $x\mapsto b\cdot F(x)\in\mathbb{F}_2$.

On a $\mathcal{C}_F$ un code $[2^n, 2n+1]$ de matrice génératrice $G_F = \begin{bmatrix} & R(1,n) & \\ F(0) & \cdots & F(1\ldots1) \end{bmatrix}$.

$$(\underbrace{\varepsilon}_{\in\mathbb{F}_2} \mid \underbrace{a}_{\in\mathbb{F}_2^n} \mid \underbrace{b}_{\in\mathbb{F}_2^n}) \begin{bmatrix} 1 & \cdots & 1 \\ & \underset{x_1}{} & \\ & \vdots & \\ & \underset{x_n}{} & \\ F(0) & \cdots & F(1\ldots1) \end{bmatrix} = (\varepsilon + a\cdot x + b\cdot F(x))_{x\in\mathbb{F}_2^n}$$

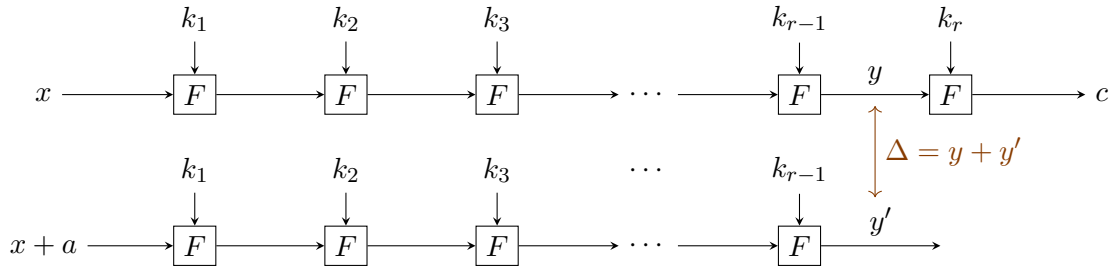
Donc $d_{\min}(\mathcal{C}_F) = 2^{n-1} - \frac{1}{2}\mathcal{L}(F)$ et $wt(f) = 2^{n-1} - \frac{1}{2}\mathcal{E}(f)$.

Proposition 8.22

Pour le code $\mathcal{C}_F$, $d_{\min} \leqslant 2^{n-1} - 2^{\frac{n-1}{2}}$, donc $\mathcal{L}(F) \geqslant 2^{\frac{n+1}{2}}$.
Il y a égalité pour les fonctions presque courbes (almost bent), qui existent uniquement si n est impair.

Si n est pair, on connaît des fonctions F telles que $\mathcal{L}(F) = 2^{\frac{n}{2}+1}$. Par exemple $x \mapsto x^{-1}$ sur $\mathbb{F}_{2^n}$, qui est utilisé dans l'AES.

8.3 Cryptanalyse différentielle



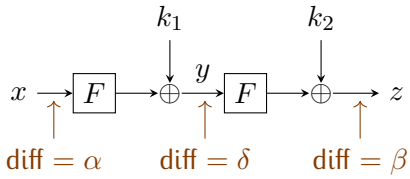
$a \neq 0$ fixé

On a $p = \mathbb{P}_{x,k}[G_k(x+a) + G_k(x) = b]$.
On va chercher un couple (a, b) pour avoir $p \gg \frac{1}{2^n-1}$.

Définition 8.23 Complexité en données

$N = O\left(\frac{1}{p}\right)$ dans les couples clairs-chiffrés choisis.

Différentielle sur deux tours



$$\mathbb{P}_{x,k_1,k_2}[\Delta z = \beta \mid \Delta x = \alpha] = \sum_{\delta} \underbrace{\mathbb{P}_{x,k_1}[\delta z = \beta \mid \Delta x = \alpha, \Delta y = \delta]}_{p'} \times \mathbb{P}_{x,k_1}[\delta y = \delta \mid \Delta x = \alpha]$$

Hypothèse

On fait l'hypothèse que le chiffrement est markovien, i.e. la suite des différences après chaque tour Δx_i est une chaîne de Markov, i.e. la proba du tour $i+1$ ne dépend que du tour i.

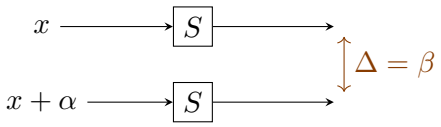
$$p' = \mathbb{P}_{x,k_1}[F(\underbrace{k_1 + F(x+\alpha)}_{y+\delta}) + F(\underbrace{k_1 + F(x)}_{y}) = \beta \mid \underbrace{F(x+\alpha) + k_1}_{y+\delta} + \underbrace{F(x) + k_1}_{y} = \delta]$$
$$= \mathbb{P}_y[F(y+\delta) + F(y) = \beta]$$

Donc

$$\mathbb{P}_{x,k_1,k_2}[\Delta z = \beta \mid \Delta x = \alpha] = \sum_{\delta} \underbrace{\mathbb{P}_y[\delta z = \beta \mid \Delta y = \delta] \times \mathbb{P}_x[\delta y = \delta \mid \Delta x = \alpha]}_{\text{chemin différentiel } (\alpha,\delta,\beta)}.$$

Différentielle sur un tour

Pour tout x on a $L(x+a) + L(x) = L(a)$. On a une boîte $S : \mathbb{F}_2^m \to \mathbb{F}_2^m$ et on calcule $\delta_S(\alpha,\beta) = \#\{x \in \mathbb{F}_2^m \mid S(x+a) + S(x) = \beta\}$, qui est toujours paire.



Définition 8.24 **Uniformité différentielle de S**

$$\delta(S) = \max_{\substack{\alpha \neq 0 \\ \beta}} \delta_S(\alpha,\beta)$$

Pour ça on calcule la table DDT

	$\beta \in \mathbb{F}_2^m$
$\alpha \in \mathbb{F}_2^m$	$\delta_S(\alpha,\beta)$

Proposition 8.25

Pour $S : \mathbb{F}_2^m \to \mathbb{F}_2^m$, on a $\delta(S) \geqslant 2$.
Il y a égalité pour les fonctions presque parfaitement non-linéaires (almost perfect nonlinear, APN).

Avec $F : \mathbb{F}_2^n \to \mathbb{F}_2^m$, on a $\delta(F) \geqslant 2^{n-m}$, avec égalité possible si $m \leqslant \frac{n}{2}$ pour les fonctions parfaitement non-linéaires.

Pour $\mathcal{C}_F$, on a $G_F = \begin{bmatrix} & R(1,n) & \\ F(0) & \cdots & F(1\ldots1) \end{bmatrix} = \begin{bmatrix} 1 & \cdots & 1 \\ & \underset{x_1}{} & \\ & \vdots & \\ & \underset{x_n}{} & \\ F(0) & \cdots & F(1\ldots1) \end{bmatrix}$.

Les mots de poids pair de $\mathcal{C}_F^\perp$ sont les colonnes de G qui sommes à zéro. On peut démontrer que $d_{\min}(\mathcal{C}_F^\perp) \leqslant 6$.
On n'a jamais $d_{\min}(\mathcal{C}_F^\perp) = 2$.
On a $d_{\min}(\mathcal{C}_F^\perp) = 4$ ssi $\exists x_1,x_2,x_3,x_4, \begin{cases} x_1+x_2+x_3+x_4 = 0 \\ F(x_1)+F(x_2)+F(x_3)+F(x_4) = 0 \end{cases}$, i.e., avec $a = x_1 + x_2$, $F(x_1) + F(x_1+a) = F(x_3) + F(x_3+a)$.
Pour $b = F(x_1)+F(x_1+a)$, on a $\{x_1, x_1+a, x_3, x_3+a\} \subseteq \{x \mid F(x+a)+F(x) = b\}$ donc $\delta_F(a,b) \geqslant 4$.

Proposition 8.26

F est APN ssi $d_{\min}(\mathcal{C}_F^\perp) = 6$.

Proposition 8.27 **Transformée de MacWilliams**

Si $d_{\min}(\mathcal{C}_F) = 2^{n-1} - 2^{\frac{n-1}{2}}$ alors $d_{\min}(\mathcal{C}_F^\perp) = 6$, i.e. si F est AB alors F est APN. Mais ce n'est possible que si n est impair.

Si $\gcd(s, 2^m - 1) = 1$ alors $F(x) = x^s$ est bijective sur $\mathbb{F}_{2^m}$. On a

$$G_F = \left[\begin{array}{c|ccc|ccc} 1 & \cdots & 1 & \cdots & 1 \\ \hline 0 & & & & \\ \vdots & & x & & \\ 0 & & & & \\ \hline 0 & & & & \\ \vdots & & x^s & & \\ 0 & & & & \end{array}\right]$$

C'est une matrice de parité d'un code cyclique à deux zéros α et α^s, donc on recherche des codes cycliques à deux zéros avec $d_{\min}(\mathcal{C})$ et $d_{\min}(\mathcal{C}^\perp)$ grandes.

— Le cas optimal est quand F est presque courbe, donc pour m impair.
  — [Gold 68] choisit $s = 2^i + 1$ avec $\gcd(i, m) = 1$ (par exemple $x \mapsto x^3$ sur $\mathbb{F}_{2^m}$).
  — [Kasami 71] choisit $s = 2^{2i} - 2^i + 1$ avec $\gcd(i, m) = 1$.
  — [Lachaud-Wolfmann 90] choisit $s = 2^m - 2$.

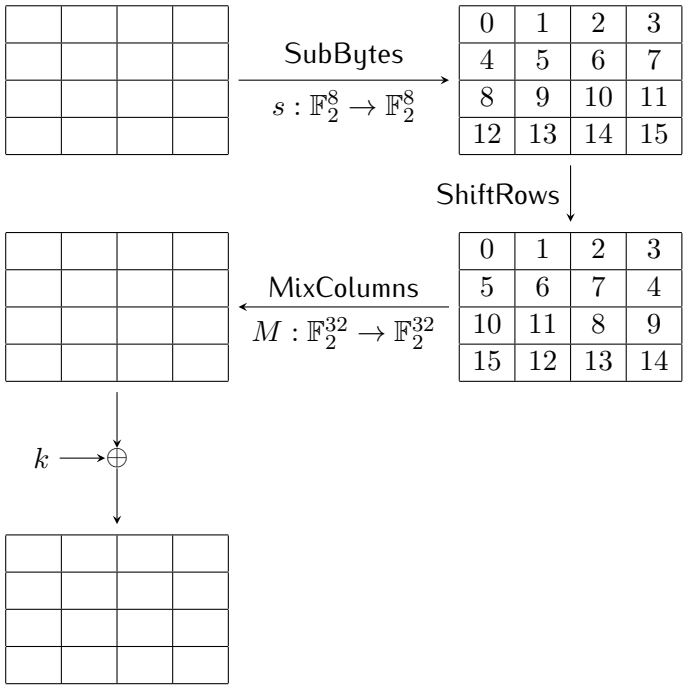
— En général on préfère avoir m pair, pour travailler sur des logiciels.
  Avec $\delta(a, b) = \#\{x \in \mathbb{F}_{2^m} \mid F(x + a) + F(x) = b\}$ et $\delta(s) = \max_{a,b\neq 0} \delta(a, b)$, on a pour $a \neq 0$ et $x \in \mathbb{F}_{2^m}$, on peut écrire $x = ay$. Alors $(x+a)^s+x^s = b$ ssi $(y+1)^s+y^s = \frac{b}{a^s}$. Donc $\delta(a, b) = \delta(1, \frac{b}{a^s})$.
  Pour $s = 2^m - 2$ l'équation devient $(y + 1)^{2^m-2} + y^{2^m-2} = c$, avec $c = \frac{b}{a^s}$.
  Pour $y = 0$ ou $y = 1$ on a des solutions ssi $c = 1$. Sinon on réécrit en

$$\underbrace{y(y+1)^{2^m-1}}_{=1} + \underbrace{(y+1)y^{2^m-1}}_{=1} = c(y+1)y,$$

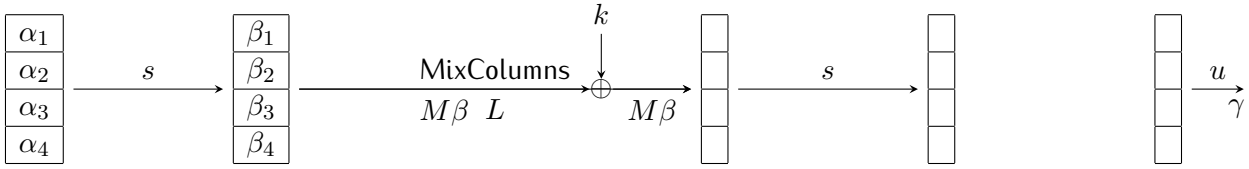
  soit en $1 = c(y + 1)y$.
  Donc $\delta(1, c) = \#\{y^2 + y = \frac{1}{c}\} \leqslant 2$.
  Si $c = 1$ on a $\delta(1, 1) = 2 + \#\{y \neq 0, 1 \mid y^2 + y + 1\} = \begin{cases} 2 & \text{si m impair} \\ 4 & \text{si m pair} \end{cases}$.
  Avec Tr : $\begin{array}{ccc} \mathbb{F}_{2^m} & \longrightarrow & \mathbb{F}_2 \\ x & \longmapsto & \sum_{i=0}^{n-1} x^{2^i} \end{array}$, on a $\mathrm{Tr}(y^2 + y + 1) = 0$ et $\mathrm{Tr}(1) = 0 = m \mod 2$.
  Donc pour m pair si on prend $F(x) = x^s$ avec $s = 2^m - 2$, $\delta(F) = 4$, $\mathcal{L}(F) = 2^{\frac{m}{2}+1}$.
  Si $\gcd(s, 2^m - 1) = 1$ alors $\delta(s) \geqslant 4$. Si $F : \mathbb{F}_2^4 \to \mathbb{F}_2^4$ est bijective, alors $\delta(s) \geqslant 4$.
  — [Dillon 2009] montre que $F : \mathbb{F}_{2^6} \to \mathbb{F}_{2^6}$ est bijective et APN. C'est la seule connue.

MixColumns

On regarde sur deux tours.



Sans la dernière f linéaire on a la concaténation de quatre copies de la même fonction $\mathbb{F}_2^{32} \to \mathbb{F}_2^{32}$.



$$\begin{aligned} p(\alpha,\beta,\gamma) &= \mathbb{P}_{x,k}[\Delta u = \gamma \text{ et } \Delta y = \beta \mid \Delta x = \alpha] \\ &= \mathbb{P}_x[\Delta y = \beta \mid \Delta x = \alpha] \underbrace{\mathbb{P}_y[\Delta u = \gamma \mid \Delta y = \beta]}_{\mathbb{P}_z[\Delta u = \gamma \mid \Delta z = M\beta]} \\ &\leqslant \left(\frac{\delta(s)}{2^s}\right)^{wt_8(\beta)} \left(\frac{\delta(s)}{2^8}\right)^{wt_8(M\beta)} \\ &= \left(\frac{\delta(s)}{2^8}\right)^{wt_8(\beta)+wt_8(M\beta)} \end{aligned}$$

Dans l'AES on a $\delta(s) = 4$. On maximise $\min_{\beta\neq 0}[wt_9(\beta) + wt_8(M\beta)]$.

Définition 8.28 — Branch number différentiel

Soit L une fonction linéaire de $(\mathbb{F}_2^m)^n \to (\mathbb{F}_2^m)^n$.
Le branch number différentiel de L est $b = \min_{\beta\neq 0}(wt_m(\beta) + wt_m(L(\beta)))$.

Avec $\mathcal{C} = \{(x, L(x)) \mid x \in (\mathbb{F}_2^m)^n\}$, $\mathcal{C}$ est de type $[2n, n, b]$.

Proposition 8.29 — Borne de Singleton

$b \leqslant 2n - n + 1 = n + 1$, avec égalité ssi le code est MDS.

Dans l'AES on utilise un code MDS $\mathcal{C}[8, 4, 5]$ sur $\mathbb{F}_{2^8}$ avec $G = \left[\ \mathrm{Id}_4 \mid M\ \right]$, où M est une matrice de MixColumns.
On a alors $p(\alpha, \beta, \gamma) \leqslant 2^{-30}$. Avec dix tours on obtient une borne de l'ordre de 2^{-150}.

Cryptanalyse linéaire

Avec $F : \mathbb{F}_2^{32} \to \mathbb{F}_2^{32}$ (deux tours sur une diagonale) on a pour $a, b \neq 0$

$$\mathcal{E}^2(b \cdot F + \varphi_a) \leqslant \left(\frac{\mathcal{L}(s)}{2^8}\right)^{2(wt_8(\beta)+w_8({}^tM\beta))}$$

Définition 8.30 **Branch number linéaire**

Soit L une fonction linéaire de $(\mathbb{F}_2^m)^n \to (\mathbb{F}_2^m)^n$.
Le branch number linéaire de L est $b' = \min_{\beta \neq 0}(wt_m(\beta) + wt_m({}^tL(\beta)))$.

On a $b' = d_{\min}(\mathcal{C}^\perp)$.

Pour simplifier les multiplications on prend une matrice M plus simples. On prend des matrices compagnon :

$$M' = \left[\begin{array}{ccccc} 0 & 1 & 0 & \cdots & 0 \\ \vdots & & \ddots & \vdots & \\ \vdots & & & \ddots & 0 \\ 0 & & & & 1 \\ \hline a_0 & & & & a_{n-1} \end{array}\right]$$

et

$$M' \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} x_2 \\ \vdots \\ x_n \\ \sum_{i=1}^n a_i x_i \end{pmatrix}.$$

Et en hardware et en software ça s'implémente bien.

Troisième partie

Thomas Debris-Alazard

Pour faire de la cryptographie à base de codes on a besoin d'un problème difficile, et on va prendre le problème de décodage.

9 | Le décodage, un problème diffcile

Problème 1

Soit $G \in \mathbb{F}_q^{k \times n}$ et $t \in [\![0, n]\!]$.
Donné $y = mG + e$ où $m \in \mathbb{F}_q^k$ et $w_H(e) = t$, retrouver e (ou mG).

Il y a une forme équivalente à ce problème.

Problème 2

Soit $H \in \mathbb{F}_q^{(n-k)\times n}$ tel que $\mathrm{rg}(H) = n - k$ et $t \in [\![0, n]\!]$.
Donné $He^\top$ où $w_H(e) = t$, retrouver e.

Dans ce cas, le code est bien $\mathcal{C} = \{x \in \mathbb{F}_q^n \mid Hx^\top = 0\}$.

Résultat 9.1

Ces deux problèmes sont les mêmes : savoir résoudre l'un en temps T équivaut à savoir résoudre l'autre en temps $\sim T$.

Preuve.
Si on a $y = mG + e$, on peut calculer $H \in \mathbb{F}_q^{(n-k)\times n}$ tel que $HG^\top = (0)$ en $O(n^3)$ avec un pivot de Gauss. Alors $Hy^\top = He^\top$.

Si on a $s^\top = He^\top$, on peut trouver un y tel que $Hy^\top = He^\top$, et on revient au problème 1 en calculant G.

On va plutôt raisonner à partir du problème 2 parce qu'on ne veut pas s'embêter avec le mot de code de base.

9.1 Qu'est-ce qu'un problème difficile ?

Pour pouvoir dire que le problème de décodage est **NP**-complet, il faut l'écrire sous forme de problème de décision :

Problème 3

Input : $s \in \mathbb{F}_q^{n-k}$, $H \in \mathbb{F}_q^{(n-k)\times n}$, t
Question : $\exists e \in \mathbb{F}_q^n$ tel que $He^\top = s^\top$ et $w_h(e) = t$?

Proposition 9.2

Le problème 3 est **NP**-complet.

Preuve.
On va faire le cas $q = 2$.

Problème 3DM

Input : T fini, $U \subseteq T^3$
Question : $\exists V \subseteq U$ tel que $\#V = \#T, \forall (x_1, y_1, z_1) \neq (x_2, y_2, z_2) \in V, x_1 \neq x_2, y_1 \neq y_2, z_1 \neq z_2$?

3DM peut s'écrire sous forme de matrice d'incidence.

Exemple 9.3

Avec $T = \{1, 2, 3\}$ et $U = \{u_1, u_2, u_3, u_4, u_5\}$ où $u_1 = (1, 1, 2)$, $u_2 = (2, 3, 1)$, $u_3 = (1, 2, 3)$, $u_4 = (3, 1, 2)$, $u_5 = (2, 2, 2)$.

	112	231	123	312	222
1	1	0	1	0	0
2	0	1	0	0	1
3	0	0	0	1	0
1	1	0	0	1	0
2	0	0	1	0	1
3	0	1	0	0	0
1	0	1	0	0	0
2	1	0	0	1	1
3	0	0	1	0	0

U, T admet une solution ssi il existe $e \in \mathbb{F}_3^{|U|\times|T|}$ tel que $w_H(e) = \#T$ et $H_{\text{3DM}} e^\top = \begin{pmatrix} 1 \\ \vdots \\ 1 \end{pmatrix}$.

9.2 Problème de McEliece

Décoder de manière générale est difficile. On ajoute des structures aux codes pour avoir des algorithmes

efficaces de décodage. Ces structures peuvent alors servir de trapdoor.

Alice, qui a la clé secrète sk, se donne H une matrice de parité d'un code qu'elle sait décoder à distance $\leqslant t$.
Elle rend public (H, t).
Bob veut chiffrer m à Alice. Il y a en encodage public φ tel que $\varphi(m) = e$ et $w_H(e) = t$.
Bob envoie $He^\top$ à Alice, qui peut retrouver e par décodage et retrouver m avec φ^{-1}.

Remarque 9.4

Ce chiffrement est celui de Niederreiter. McEliece utilise des matrices génératrices.

Exercice 9.5 **Pour la semaine prochaine**

Écrire ce schéma de chiffrement avec des matrices génératrices.

En pratique il y a des codes qu'on sait décoder.
Par exemple si $q \geqslant n$, avec $x = (x_1, ..., x_n)$ et $x_i \neq x_j$, et $y = (y_1, ..., y_n)$ et $y_i \neq 0$, on a le code de Reed-Solomon généralisé

$$\mathcal{C} = \mathrm{GRS}_k(x, y) = \{(y_1 f(x_1), ..., y_n f(x_n)) \mid f \in \mathbb{F}_q[X], \deg(f) < k)\}.$$

$\mathcal{C}$ se décode jusqu'à distance $\left\lfloor \frac{n-k}{2} \right\rfloor$ *si on a la connaissance des* x_i *et* y_i.
Le secret d'Alice est les x_i et y_i.

$$H = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ x_1 & x_2 & \cdots & x_n \\ \vdots & \vdots & & \vdots \\ x_1^{n-k-1} & x_2^{n-k-1} & \cdots & x_n^{n-k-1} \end{pmatrix} \begin{pmatrix} y'_1 & & \\ & \ddots & \\ & & y'_n \end{pmatrix}$$

où $y'_i = \frac{1}{y_i \prod_{j \neq i}(x_i - x_j)}$.

Alice ne doit pas rendre public H car cela révèle les x_i et y_i. En revanche, si elle tire aléaoirement une matrice $S \in \mathbb{F}_q^{(n-k)\times n}$ inversible, alors SH est une matrice de parité de $\mathcal{C}$ choisie aléatoirement. Alice rend alors public SH.
Malheureusement c'est cassé.

9.3 Le décodage en cryptographie

9.3.1 Difficulté en moyenne

t désigne ici une distance de décodage, n la longueur du code et k sa dimension. On note $\tau = t/n$ et $R = k/n$. On a $\tau : \mathbb{N} \to]0, 1[$ et $R : \mathbb{N} \to]0, 1[$, τ et R dépendent de n.

Problème $\mathrm{DP}(q, n, \tau, R)$

Entrée : $H \in \mathbb{F}_q^{(n-k)\times n}$ tirée aléatoirement, $s^\top = Hx^\top$ avec x tiré uniformément parmi les mots de poids $t = \tau \times n$
Sortie : e tel que $w_H(e) = t$ et $He^\top = s^\top$

On a un $y = c + e'$ en entrée et on veut récupérer un e tel que $w_H(e) = t$ et $y - e \in \mathcal{C}$.

Remarque 9.6

Selon t on va retrouver le bon c, mais ce n'est pas forcément ce que l'on veut en cryptographie.

Question 9.7

Pourquoi a-t-on des distributions en entrée du problème ?

Soit $\mathcal{A}$ un algorithme avec en entrée $H_0 \in \mathbb{F}_q^{(n-k)\times n}$ et $s_0 = H_0 x_0^\top$ où $w_H(x_0) = t$, tel que

$$\mathcal{A}(H_0, s_0) \longmapsto \begin{cases} e & \text{tel que } w_H(e) = t \text{ et } H_0 e^\top = s_0^\top \\ \perp \end{cases}$$

et $\mathcal{A}$ fonctionne en temps T.
On pose

$$\varepsilon = \mathbb{P}_{H, s=Hx^\top, w}(\mathcal{A}(H, s, w) = e, w_H(e) = t \text{ et } He^\top = s^\top)$$

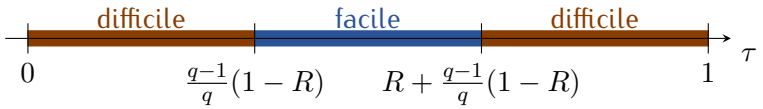
où $H \xleftarrow{\$} \mathbb{F}_q^{(n-k)\times n}$, $x \xleftarrow{\$}$mots de poids t, w aléa interne de $\mathcal{A}$.

ε est la probabilité moyenne de succès de $\mathcal{A}$.
Par formule des probabilités totales,

$$\begin{aligned}\varepsilon &= \sum_{\substack{H_0 \in \mathbb{F}_q^{(n-k)\times n} \\ x_0 \in \mathbb{F}_q^n, w_H(x_0)=t}} \mathbb{P}_{H, s=Hx^\top, w}(\mathcal{A}(H, s, w) = e\ldots \mid H = H_0, x = x_0)\mathbb{P}(H = H_0, x = x_0) \\ &= \frac{1}{q^{(n-k)\times n}} \times \frac{1}{\binom{n}{k}(q-1)^t} \sum_{\substack{H_0 \\ x_0}} \mathbb{P}_w(\mathcal{A}(H_0, s_0, w) = e).\end{aligned}$$

On dit alors que $\mathcal{A}$ résout le problème en temps moyen T/ε.

C'est la contraire avec t qui rend le problème difficile. Avec $\tau(n) = \frac{\log(n)}{n}$ on a un nombre d'erreurs possibles polynomial en n et donc on peut retrouver avec force brute.
Les meilleurs décodeurs génériques sont de complexité moyenne exponentielle en $\tau \times n$. Mais pour certains τ on peut avoir mieux.



Si on oublie la constante de poids, on a un système linéaire à résoudre, de $(n-k)$ équations à n inconnues. On va fixer k inconnues comme ça nous arrange.
On coupe H en deux : $H = (\underset{k}{A} \mid \underset{n-k}{B})$ avec $B \in \mathbb{F}_q^{n-k}$ inversible. On découpe $e = (\underset{k}{e_1} \mid \underset{n-k}{e_2})$. Alors $He^\top = s^\top$ ssi $Ae_1^\top + Be_2^\top = s^\top$ ssi $e_2^\top = B^{-1}(s^\top - Ae_1^\top)$.
Alors $\mathbb{E}(w_H(e_2)) = \frac{q-1}{q}(n-k)$, donc $\mathbb{P}(w_H(e_2) \leqslant (1-\varepsilon)\frac{q-1}{q}(n-k))$ est exponentiellement faible, de même pour $\mathbb{P}(w_H(e_2) \geqslant (1+\varepsilon)\frac{q-1}{q}(n-k))$.

9.4 Chiffrement d'Alekhnovich

Génération de clé

On a $\mathcal{C}$ un code aléatoire (de longueur n et dimension k).
$e_{\text{sk}} \xleftarrow{\$}$ mots de poids t
$c \xleftarrow{\$} \mathcal{C}$
$\begin{cases} \text{pk} = (\mathcal{C}, c + e_{\text{sk}}) \\ \text{sk} = e_{\text{sk}} \end{cases}$

Chiffrement

On veut chiffrer un bit $b \in \{0, 1\}$.

— Si $b = 1$, on prend $u \xleftarrow{\$} \mathbb{F}_2^n$

— Si $b = 0$, on tire $c^* \xleftarrow{\$} (\mathrm{Span}(\mathcal{C}, c + e_{\mathrm{sk}}))^{\perp}$ et $e^* \xleftarrow{\$}$ mots de poids t, et on prend $c^* + e^*$

Déchiffrement

On calcule le produit scalaire entre e_{sk} et le mot reçu, on trouve 0 avec grande probabilité si $b = 0$, et un résultat uniforme si $b = 1$.

Index des définitions

Index des résultats