Chapitre 3 Codes et codages linéaires.

Si l'alphabet $A = \mathbb{K}$ est un *corps de Galois* (corps fini) de cardinal q noté $\mathbb{F}_q$ tel que $q = p^r$ où p entier premier représentant la caractéristique de $\mathbb{K}$, alors $\mathbb{K}^n$ est un $\mathbb{K}$-espace vectoriel de dimension n pour les lois habituelles (l'addition et la multiplication par un scalaire), dans la pratique, on prend $\mathbb{K} = \mathbb{F}_2 = \{0,1\}$.

Dans tous le chapitre $\mathbb{K} = \mathbb{F}_q$ tel que $q = p^r$, est un corps de Galois.

3.1 Définitions et propriétés.

Définition 3.1.1.

On appelle *code linéaire* (ou *code q-aire*) C de *longueur n* et de *dimension k* sur $\mathbb{K}$, tout sous-espace vectoriel du $\mathbb{K}$-espace vectoriel $\mathbb{K}^{\boldsymbol{n}}$, de dimension k. Si la distance de C est *d*, on le note *C*(*n*, *k*, *d*) où *n, k, d* sont dits *paramètres* du code *C*

Remarque 3.1.1.

1- C est un code linéaire, si et seulement si, pour tous $x_1, x_2 \in$ C, $\alpha_1, \alpha_2 \in \mathbb{K}$, $\alpha_1 x_1 + \alpha_2 x_2 \in$ C

2- Une application $\phi: \mathbb{K}^k \longrightarrow \mathbb{K}^n$ sur l'alphabet $\mathbb{K}$ est dite *codage linéaire*, si et seulement si l'application ϕ est une application linéaire injective. Le code linéaire associé à l'application ϕ est son image. C.à.d. C= Imϕ=$\{\phi(x) / x \in \mathbb{K}^k\}$.

Exemples 3.1.1.

1- $\{0\}$ et $\mathbb{K}^{\boldsymbol{n}}$ sont des codes linéaires dits *codes triviaux*.

2- $\phi: \mathbb{K}^k \longrightarrow \mathbb{K}^{k+1}$ *codage par bit de parité*.

$x = (x_1, \dots, x_k) \longmapsto \phi(x) = (x_1, \dots, x_k, \sum_{i=1}^{k} x_i)$ est codage linéaire car l'application ϕ est linéaire.

3- $\phi : \mathbb{K} \longrightarrow \mathbb{K}^n$ *codage à répétition.*

$x = x_1 \longmapsto c = \phi(x) = (x_1, \dots, x_1)$ est un codage linéaire.

4- Le code C=$\{(x_1, x_2, x_3, x_4, x_1 + x_2 + x_3,\ x_1 + x_2 + x_4, x_2 + x_3 + x_4) / (x_1, x_2, x_3, x_4) \in {F_2}^4\}$ est un code linéaire car c'est un sous-espace vectoriel de ${F_2}^4$, engendré par la base:

B =$\{L_1$=(1, 0, 0, 0, 1, 1, 0), L_2=(0, 1, 0, 0, 1, 1, 1), L_3=(0, 0, 1, 0, 1, 0, 1), L_4=(0, 0, 0, 1, 0, 1, 1)$\}$.

Définition 3.1.2.

Le *poids* d'un élément $x = (x_1, \dots, x_n) \in \mathbb{K}^n$ noté $\boldsymbol{w(x)}$, est le nombre de ses composantes non nulles.

$w(x) = card\{i \in \{1,2, \dots, n\}: x_i \neq 0\}$.

Exemple 3.1.2. Soit l'alphabet A =$\mathbb{F}_2$ et $x = (1,0,1,0) \in \mathbb{F}_2^4$, $w(x) = 2$

Proposition 3.1.1.

Pour tous $x, y \in \mathbb{K}^{\boldsymbol{n}}, \lambda \in \mathbb{K}$ on a:

$i)$ $d(x,y) = w(x-y)$

$ii)$ $w(x) = d(x,0)$

$iii)$ $w(x) = 0 \Leftrightarrow x = 0$

$iv)\ w(\lambda x) = w(x),, \forall \lambda \neq 0$

$v)\ \ w(x+y) \leq w(x) + w(y)$

Preuve.

Pour tous $x, y\ \in \mathbb{K}^{\boldsymbol{n}}, \lambda \in \mathbb{K}$ on a:

i) $d(x,y) = card\{i \in \{1,\dots,n\},\ x_i \neq y_i\} = card\{i \in \{1,\dots,n\},\ x_i - y_i \neq 0\}$

$= d(x-y,0) = w(x-y).$

ii) $w(x) = d(x,0)$

On $a : d(x,0) = \ card\{i \in \{1,\dots,n\}/x_i \neq 0\} = w(x).$

iii) $(w(x) = 0) \Leftrightarrow (d(x,0) = 0) \Leftrightarrow (x = 0).$

iv) $\lambda \in\ \mathbb{K} : \lambda \neq 0 : w(\lambda x) = w(x).$

$w(\lambda x)$=d $(\lambda x, 0) = card\{i \in \{1,\dots,n\}/\lambda x_i \neq 0\}$ = $card\{i \in \{1,\dots,n\}/\lambda \neq 0 \wedge x_i \neq 0\}$

= $card\{i \in \{1,\dots,n\}/x_i \neq 0\} = w(x).$

v) $w(x+y) \leq w(x) + w(y)$.

$w(x+y) = d(x+y,0) \leq d(x+y,x) + d(x,0)$ (d est une distance)

Donc : $w(x+y) \leq d(x+y-x,0) + d(x,0) = d(y,0) + d(x,0) = w(x) + w(y)$(d'après i).□

Définition 3.1.3.

On appelle *poids minimum* d'un code linéaire C(***n, k***) le plus petit poids des mots non nul du code C et on le not P_{min}

Exemple 3.1.3.

C= {00000, 01111, 11000,10111} est un code linéaire de longueur 5 et de dimension 2 sur $\mathbb{F}_2$ et

$P_{\boldsymbol{min}}$(C)= $min\{4,2\} = 2.$

Proposition 3.1.2.

Si C est un code linéaire, l'ensemble des distances entre mots de C est l'ensemble des poids des mots de C.

Preuve. D= {d(*x, y*) / *x, y*∈ C}= {w(*x-y*) / *x, y*∈ C}= {w(*z*) / *z*= *x-y*∈ C}, *z* ∈ C car C est un sous espace vectoriel de $\mathbb{K}$. □

Conséquence 3.1.1.

La distance minimale d'un code linéaire est égale au poids minimum des mots non nuls du code,

P_{min}= d_{min}.

Chercher la distance minimale d'un code en calculant le poids minimum des mots non nul est plus facile (car moins long) que chercher la plus petite distance entre tous les mots du code deux à deux distincts.

3.1.1 Matrice génératrice.

Définition 3.1.4. Matrice génératrice.

Une *matrice génératrice* d'un code linéaire C(n, k) sur le corps fini $\mathbb{K}$ est une matrice de type $k \times n$ à coefficients dans $\mathbb{K}$, dont les lignes forment une base de C.

Exemple 3.1.4. $G=\begin{pmatrix}1 & 0 & 1 & 1 & 1\\ 0 & 1 & 1 & 1 & 1\end{pmatrix}$ est une matrice génératrice du code linéaire C(5,2)

C= {00000, 01111, 11000,10111} sur $\mathbb{F}_2$. Car les vecteurs lignes L_1=10111 et L_2 = 01111 sont libres est donc forment une base de C car on sait que la dimension de C est k=2.

Proposition 3.1.3.

Si G est une matrice génératrice d'un code linéaire C(n, k) sur $\mathbb{K}$, alors :

Toute matrice génératrice de C est de la forme $A\times G$, ou A est une matrice carrée inversible d'ordre k sur $\mathbb{K}$ (ou encore A est une matrice carrée de rang k sur).

Preuve. $soit\ G' = A.G$, qui est une matrice de type $k \times n$. G' est une matrice génératrice de , si et seulement si, $rg(G') = k$?

$rg(G') = rg(A.G) = min(rg(A), rg(G)) = min\ (k, k) = k$. Donc G'=A . G est une matrice génératrice de C. □

Exemple 3.1.5.

$G=\begin{pmatrix}1 & 0 & 1 & 1 & 1\\ 0 & 1 & 1 & 1 & 1\end{pmatrix}$ une matrice génératrice du code C sur le corps $\mathbb{F}_2$ dans l'exemple précèdent alors les matrices génératrices possible de C sont:

G_0=I_2.G=G, $G_1=\begin{pmatrix}1 & 1\\ 0 & 1\end{pmatrix}.\begin{pmatrix}1 & 0 & 1 & 1 & 1\\ 0 & 1 & 1 & 1 & 1\end{pmatrix}=\begin{pmatrix}1 & 1 & 0 & 0 & 0\\ 0 & 1 & 1 & 1 & 1\end{pmatrix}$ et $G_2=\begin{pmatrix}0 & 1\\ 1 & 1\end{pmatrix}.\begin{pmatrix}1 & 0 & 1 & 1 & 1\\ 0 & 1 & 1 & 1 & 1\end{pmatrix}$

$=\begin{pmatrix}0 & 1 & 1 & 1 & 1\\ 1 & 1 & 0 & 0 & 0\end{pmatrix}$ $G_3=\begin{pmatrix}1 & 0\\ 1 & 1\end{pmatrix}.G=\begin{pmatrix}1 & 0 & 1 & 1 & 1\\ 1 & 1 & 0 & 0 & 0\end{pmatrix}$, $G_4=\begin{pmatrix}1 & 1\\ 1 & 0\end{pmatrix}.G=\begin{pmatrix}1 & 1 & 0 & 0 & 0\\ 1 & 0 & 1 & 1 & 1\end{pmatrix}$ et

$G_5=\begin{pmatrix}0 & 1\\ 1 & 0\end{pmatrix}.G=\begin{pmatrix}0 & 1 & 1 & 1 & 1\\ 1 & 0 & 1 & 1 & 1\end{pmatrix}$.

3.1.2 Construction d'un code linéaire.

Proposition 3.1.4.

Si G est la matrice génératrice d'un code linéaire C(n, k) sur $\mathbb{K}$ alors: Le code C est le sous-espace de $\mathbb{K}^n$ des mots de la forme : c= x.G avec $x=(x_{1,\dots,}x_k) \in \mathbb{K}^k$.

Preuve. Comme *G* est une matrice de type (*k*, *n*) et de rang *k* (car les k lignes de *G* sont libres) alors ${}^tG$ est une matrice de type (*n*, *k*) et de rang k il existe donc une application linéaire ϕ de $\mathbb{K}^k$dans $\mathbb{K}^n$ tel que ${}^tG$ est la matrice de ϕ dans les bases canoniques de $\mathbb{K}^k$ et $\mathbb{K}^n$ et on a :

dim(kerϕ) = *k*-rg(ϕ) = *k*-rg(*G*) = *k*-*k* = 0 d'où Ker(ϕ) = {0}et ϕ est injective donc c'est un codage linéaire dont l'image est C et on a : $C=\{\phi(x) / x=(x_{1,\dots,}x_k) \in \mathbb{K}^k\}= \{x.\ {}^t({}^tG) / x=(x_{1,\dots,}x_k) \in \mathbb{K}^k\}= \{x.G / x=(x_{1,\dots,}x_k) \in \mathbb{K}^k\}$. □

Remarque 3.1.2. Si *G* est une matrice génératrice d'un code linéaire C (*n*, *k*) sur $\mathbb{K}$, alors les mots de C sont toutes les combinaisons linéaires des lignes $L_1, L_2, \dots, L_n$ de *G*.

($i.e$: pour tout $c \in C: c = \sum_{i=1}^{k} \alpha_i L_i / \alpha_i \in \mathbb{K}$.).

Exemple 3.1.6. Soit C un code linéaire de type (2, 5) de matrice génératrice

$G=\begin{pmatrix}1 & 0 & 1 & 1 & 1\\ 1 & 1 & 1 & 1 & 0\end{pmatrix}$ alors le code linéaire associé est:

$C = \{(x_1, x_2).G / x_1, x_2 \in \{0,1\}\}= \{(x_1+x_2, x_2, x_1+x_2, x_1+x_2, x_1) / x_1, x_2 \in \{0,1\}\}$

d'où C={00000, 10111, 11110, 01001}.

Exemple 3.1.7. Considérons la matrice :$G = \begin{pmatrix}1 & 1 & 1 & 1 & 0\\ 0 & 1 & 0 & 1 & 1\\ 0 & 0 & 1 & 0 & 1\end{pmatrix}$

La matrice G est de rang 3, en effet les trois premières colonnes de G sont linéairement indépendantes, c'est la matrice génératrice d'un code linéaire C(5,3) sur $\mathbb{F}_2$={0,1} dont les mots sont: $C = \{0,\ L_1,\ L_2,\ L_3\ ,\ L_1 + L_2,\ L_1 + L_3,\ L_2 + L_3,\ L_1 + L_2 + L_3\ \}$ donc

C={ 00000, 11110,01011,00101,10101,11011,01110,10000}.

3.1.3 Codage de Hamming (7,4)

C'est un code linéaire binaire qui permet de **détecter et corriger une erreur** sur un bit dans un bloc de 7 bits.

- **4 bits** sont des bits de données (d_1 d_2 d_3 d_4).
- **3 bits** sont des bits de parité (p_1 p_2 p_3), calculés à partir des données.

Les bits de parité (p_1 p_2 p_3) sont placés aux positions puissances de 2 (1, 2, 4) c.à.d. p_1 en position 1, p_2 en position 2 et p_3 en position 4 et d_1 d_2 d_3 d_4 en position 3, 5, 6, 7 respectivement.

Le codage du mot **d = d_1 d_2 d_3 d_4** sera le mot ccode **c=p_1 p_2 d_1 p_3 d_2 d_3 d_4** tel que p_1, p_2, p_3 sont calculer comme suit :

p_1 se calcule tel que la parité des bits en position 1,3,5,7 soit paire c.a.d. $p_1=d_1+d_2+d_4$.

p_2 se calcule tel que la parité des bits en position 2,3,6,7 soit paire c.a.d. $p_2=d_1+d_3+d_4$.

p_3 se calcule tel que la parité des bits en position 1,3,5,7 soit paire c.a.d. $p_3=d_2+d_3+d_4$.

En résumé est comme suit :

Le codage du mot **d = d_1 d_2 d_3 d_4** est le mot code **c= d_1+d_2+d_4 d_1+d_3+d_4 d_1 d_2+d_3+d_4 d_2 d_3 d_4.**

Le code de Hamming(7,4) admet comme matrice génératrice la matrice G donné par :

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

3.2 Matrice génératrice normalisé et code linéaire systématique.

Dans ce qui suit $\mathbb{K}$ est un corps fini de cardinal q. La proposition ci-dessous montre comment construire un code linéaire en utilisant sa matrice génératrice.

3.2.1 Matrice génératrice normalisé

Définition 3.2.1.

Une matrice génératrice G_N d'un code linéaire $\mathrm{C}(n,k)$ est *normalisée* (*standard*) si la matrice formée par ses k premières colonnes est la matrice unité I_k.

Donc G est de la forme $(I_k|M)$ tq : $M \in M_{k,n-k}(\mathbb{K})$ est une matrice dite *de parité* ou *de contrôle*.

Exemples 3.2.1.

1. Soit C un code linéaire (5,2) sur $\mathbb{F}_2$ de matrice génératrice $G=\begin{pmatrix} 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 \end{pmatrix}$. G est une matrice génératrice normalisée du code C et C={00000, 01111, 11000, 10111}

2. $G=\begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$ est une matrice génératrice normalisée d'un code linéaire C(5,3) sur $\mathbb{F}_2$ dont les mots codes sont : C={10010, 01011, 0010, 11001, 10111, 01110, 11100, 00000}

Remarque 3.2.1.

Certains codes linéaires n'admettent pas de matrices génératrices standard, par exemple le code: C= {000, 001, 010,011} admet comme matrices génératrices de C les matrices ;

$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$

Donc C n'admet pas de matrice génératrice normalisée.

3.2.2 Codes linéaires systématiques

Définition 3.2.2. Code linéaire systématique :

Un code linéaire $\mathrm{C}(n,k)$ est dit *systématique* (ou *standard*) s'il possède une matrice génératrice normalisée G_N de la forme $(I_k|M)$ tq : $M \in M_{k,n-k}(\mathbb{K})$.

Remarque 3.2.2.

1. Quelques mathématiciens définissent la matrice génératrice normalisée par:

$G_N = (M \mid I_k)$ tq : $M \in M_{k,n-k}(\mathbb{K})$.

2. Si C est un code systématique linéaire de matrice génératrice normalisé $G_N=(I_k|M)$ alors C=$\{x.G_N / x \in \mathbb{K}^k\}=\{(x, x.M) / x \in \mathbb{K}^k\}$.

Exemple 3.2.2. Considérons le code linéaire binaire de matrice génératrice normalisée:

$G=\begin{pmatrix}1&1&1&1&0\\0&1&0&1&1\\0&0&1&0&1\end{pmatrix}$ de la forme $G=A|B)$. Comme la matrice $A=\begin{pmatrix}1&1&1\\0&1&0\\0&0&1\end{pmatrix}$ est inversible alors C est systématique de matrice génératrice normalisée $G_N=\begin{pmatrix}1&0&0&0&0\\0&1&0&1&1\\0&0&1&0&1\end{pmatrix}$ de la forme $G_N=(I_3|M)$ avec

$M=\begin{pmatrix}0&0\\1&1\\0&1\end{pmatrix}$ et donc le code C=$\{x.G_N / x \in \mathbb{F}_2{}^3\}=\{(x, x.M) / x=(x_1, x_2, x_3) \in \mathbb{F}_2{}^3\}$

C=$\{(x_1, x_2, x_3, x_2, x_2+x_3) / x_i \in \mathbb{F}_2\}$ = { 00000, 11110,01011,00101,10101,11011,01110,10000}.

Proposition 3.2.1.

Si elle existe, la matrice génératrice normalisée d'un code linéaire systématique C est unique. On l'obtient en appliquant l'algorithme de GAUSS sur une matrice génératrice quelconque de C.

Preuve. Soit $G=(A \mid B)$ ou A est une matrice carré de rang k (donc inversible), en appliquant l'Algorithme de Gauss sur les lignes de G (donc de A) pour avoir la matrice unité I_k, alors on obtient une matrice G_N (équivalente à G) de la forme $G_N =(I_k \mid B')$ qui est une matrice normalisée de C. □

Exemple 3.2.3.

Le code de Hamming binaire C de longueur n=7 et de dimension k=4, définit par:

C = $\{(x_1+x_2+x_3, x_1+x_3+x_4, x_1, x_2+x_3+x_4, x_2, x_3, x_4) / x_1, x_2, x_3, x_4 \in \{0,1\}\}$.

est un code linéaire de base B={L_1=1110000, L_2=1101100, L_3=1001010, L_4=0101001} qui admet la matrice génératrice suivante: G = $\begin{pmatrix}1&1&1&0&0&0&0\\1&1&0&1&1&0&0\\1&0&0&1&0&1&0\\0&1&0&1&0&0&1\end{pmatrix}$, c'est un code systématique de matrice normalisée G_N = $\begin{pmatrix}1&0&0&0&0&1&1\\0&1&0&0&1&0&1\\0&0&1&0&1&1&0\\0&0&0&1&1&1&1\end{pmatrix}$, ce code est de distance minimale d=3? à vérifier.

Exemple 3.2.4. Soit C un code linéaire définit par sa matrice génératrice

G = $\begin{pmatrix}1&1&1&0&0&0\\1&0&1&1&0&1\\0&1&1&1&1&0\end{pmatrix}$ de la forme $G=(A, B)$ où $A=\begin{pmatrix}1&1&1\\1&0&1\\0&1&1\end{pmatrix}$ inversible donc C est systématique on applique l'algorithme de GAUSS à G, pour passer de $G=(A, B)$ à $G'=(I_3,$

$B')$:$G=\begin{pmatrix}1&1&1&0&0&0\\1&0&1&1&0&1\\0&1&1&1&1&0\end{pmatrix} L_2 \leftarrow L_2+L_1 \sim \begin{pmatrix}1&1&1&0&0&0\\0&1&0&1&0&1\\0&1&1&1&1&0\end{pmatrix}$

$L_3 \leftarrow L_3+L_2 \sim \begin{pmatrix} 1 & 1 & 1 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} L_1 \leftarrow L_1+L_3 \sim \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix} L_1 \leftarrow L_2+L_1$

$\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$ = G_N qui est une matrice normalisée de C.

C = $\{(x_1, x_2, x_3, x_2+x_4, x_1+x_3+x_4, x_1+x_2+x_4) / x_1, x_2, x_3, x_4 \in \{0,1\}\}$.

Remarque 3.2.3.

1. Si le code linéaire C (n, k) est systématique alors pour chaque mot x= ($x_1, x_2, \ldots, x_k$) de $\mathbb{K}^k$, il existe un mot c est un seul de C de la forme : c =($x_1, x_2, \ldots, x_k, x_{k+1}, \ldots, x_n$).
2. Si la matrice génératrice G est de la forme G= (A | B) où A est inversible, alors le code associé est systématique de matrice génératrice normalisée G_N=(I_k | $A^{-1}B$).
3. Si la matrice génératrice G est de la forme G= (A | B) où A n'est pas inversible, alors le code associé ne peut être systématique, mais équivalent à un code systématique.

Théorème 3.2.1. Tout code linéaire C(n, k) est équivalent à un code linéaire systématique.

Preuve. Supposons que C ne soit pas un code systématique. Soit G une matrice génératrice du code C. Comme le rang de G est égal à k, il existe un mineur $k \times k$ non nul. Par une permutation des colonnes de G, on amène ce mineur aux k premières colonnes, on obtient ainsi une matrice génératrice d'un code linéaire systématique C' équivalent à C. □

Exemple 3.2.5. Soit le code linéaire C (3, 2) de l'Exemple 3.2.1 C= {000, 001, 010,011} qui n'est pas systématique (car il n'admet pas de matrice standard), et soit la matrice génératrice G=$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 1 & 1 \end{pmatrix}$ en faisant les permutations sur les colonnes $C_1 \leftrightarrow C_2$ et $C_2 \leftrightarrow C_3$ on obtient la matrice G_1=$\begin{pmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \end{pmatrix}$, et en appliquant l'algorithme de Gauss sur les lignes de G_1 ($L_2 \leftarrow L_2+L_1$)on obtient la matrice G_N =$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$ qui est la matrice génératrice normalisée d'un code systématique
C'= {000, 100, 010,110} équivalent à C.

Exemple 3.2.6. Soit $\mathbb{K} = \{0,1\}$ et G la matrice binaire de type (4, 6) suivante:

G = $\begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$. Le premier mineur 4×4 : $\begin{vmatrix} 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \end{vmatrix}$ est nul car la somme des deux premières colonnes est égale à la troisième par contre : $\begin{vmatrix} 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{vmatrix} \neq 0$, on en déduit que rg($G$)=4,

donc G est la matrice d'un code linéaire binaire C(6,4, d) mais à cause de la remarque du début, C

n'est pas systématique, on permute les colonnes (pour amener le mineur non nul aux 4 premières colonnes) par $\pi = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 2 & 4 & 5 & 3 & 6 \end{pmatrix}$ pour avoir G'=$\begin{pmatrix} 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}$. Puis on applique l'algorithme de Gauss sur les lignes de *G*' et on obtient $G_{N=}\begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \end{pmatrix}$ qui est la matrice génératrice d'un code systématique équivalent à C.

3.3 Bornes sur la distance minimale :

Soit C(*n, k, d*) un code linéaire sur un corps fini K de cardinal *q*. Les grandeurs q^k et d "jouent" l'une contre l'autre.

En effet, si on a un très grand nombre de mots (i.e ; *k* très grand), alors on aura une distance d faible. Alors que si on a *k* est petit, alors la distance d sera grande, permettant ainsi de détecter et de corriger plus d'erreurs.

Il va donc être nécessaire de trouver un compromis entre ces deux valeurs, afin d'avoir un nombre de mots suffisant et une distance minimale suffisamment grande pour pouvoir détecter et corriger un certain nombre d'erreurs. Pour cela, il existe des bornes qui caractérisent les grandeurs *k* et *d*. Parmi ces bornes on trouve la borne de singleton, de Griesmer et de Hamming.

3.3.1 Borne de Singleton et codes M.D.S.

Théorème et définition 3.3.1. Si d est la distance minimale d'un code linéaire $\mathrm{C}(n,k)$ alors : $d \leq n-k+1$. Cette borne est appelée *borne de singleton* du code C.

Preuve.

Méthode 1. On sait qu'un code linéaire C est équivalent à un code linéaire systématique C'et que les paramètres (*n, k, d*) des codes C et C' sont les mêmes; on peut donc faire la démonstration pour le code C'. Soit G la matrice génératrice normalisée du code C', tout mot du code C' s'écrit donc comme combinaison linéaire des lignes de *G*, le poids minimum du code est donc nécessairement inférieur au poids minimum des vecteurs composant les lignes de *G*, et à fortiori inférieur au poids maximum des vecteurs composant les lignes de *G*, les *k* premières colonnes de la matrice normalisée *G* formant la matrice identité, le poids maximum d'une ligne est donc majoré par 1+(*n-k*),donc la distance minimale d'un code linéaire est inférieur à *n-k*+1.

Méthode 2. Soit le sous espace vectoriel D={ $x = (x_1, x_2, \dots, x_n) \in \mathbb{K}^n : \forall\, i \geq d \;\; x_i = 0$}

D={ x= ($x_{1,}.., x_{d-1}, 0 \ldots, 0$) $\in \mathbb{K}^n$} alors dim(D)=*d*-1et pour tout $x \in$D-{0}: w(x)<d donc $x \notin$C et C$\cap$D={0} et on a dim(C+D)=dim(C) + dim(D)-dim(C$\cap$D)=*k*-(*d*-1) et comme C+D $\subset \mathbb{K}^n$ donc dim(C$\cap$D) $\leq n$ d'où $d \leq n$-k+1.

Cette borne permet de trouver une borne maximale sur la distance minimale *d* par rapport aux valeurs *n* et *k*. □

Définition 3.3.1. Un code linéaire $C(n, k)$ est dit *code M.D.S* (maximum distance séparable) si sa distance minimal *d* atteint la borne de singleton i.e. : $d = n - k + 1$.

Exemple 3.3.1.

le code à répétition C(*n*, *k*=1, d=*n*) est un code linéaire de matrice génératrice

G=(1 1 … 1 1), est un code M.D.S car *n*-*k*+1= *n* = *d*.

Exemple 3.3.2. Pour le code à bit de parité C(*n*, *k*=*n*-1)

ϕ: $A^{n-1} \longrightarrow A^n$

$x = (x_1, \ldots, x_k) \longmapsto \phi(x) = (x_1, \ldots, x_k, \sum_{i=1}^{k} x_i)$

C est un code M.D.S? A vérifier.

Exemple 3.3.3. Soit $G = \begin{pmatrix} 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 \end{pmatrix}$ une matrice génératrice normalisée d'un code linéaire C(*n*=5, *k*=3) sur $\mathbb{F}_2$: Est-ce que le code C est M.D.S?

3.3.2 Borne de Hamming ou d'empilement de sphères.

Définition 3.3.2. Soit Si $C(n, k, d)$ est un code linéaire sur un corps *K* tel que card(*K*)=*q* et $r \in \mathbb{N}^*$.

Pour tout $x \in K^n$ on définit B(x, *r*)={$y \in K^n$: d(x, y) $\leq r$ } *la boule de centre x* et *de rayon r* et S(x, *i*) = {$y \in K^n$: *d*(x, y) = *i* } *la sphère de centre x* et de *rayon i*.

Lemme 3.3.1. Pour tout $i \leq n$: pour tout $y \in K^n$ card(B(x, *r*)) = $\sum_{i=0}^{i=r} C_n^i\,(q-1)^i$.

Preuve. On a B(x, *r*)={$y \in K^n$: d(x, y) $\leq r$ }. Comme la distance *d* est un entier alors on peut écrire B(x, *r*)= $\bigcup_{i=0}^{i=r} S(x, i)$ tel que les S(x, *i*) sont disjointes deux à deux et donc card(B(x, *r*)) = $\sum_{i=0}^{i=r} card(S(x, i))$. Pour x et *i* fixés, calculons card($S(x, i)$) qui est le nombre des éléments $y \in K^n$: dont le nombre de composantes distinctes de celles de x est égale à *i*. Comme les mots sont de longueur *n*, il y a donc C_n^i ensembles d'indices à *i* éléments possibles.

Chaque composante admet (*q*-1) possibilités et donc card($S(x, i)$)= $C_n^i(q-1)^i$

D'où card(B(x, *r*)) =$\sum_{i=0}^{i=r} card(S(x, i))$= $\sum_{i=0}^{i=r} C_n^i\,(q-1)^i$. □

Théorème et définition 3.3.2. Si $\mathrm{C}(n,k,d)$ est un code linéaire sur un corps fini *K* tel que card(*K*)=*q* et sa capacité de correction *e*=$[\frac{d-1}{2}]$ alors, $q^{n-k} \geq \sum_{i=0}^{i=e} C_n^i\,(q-1)^i$. Cette borne est dite *borne d'empilement de sphères (ou borne de Hamming).*

Preuve. On sait que dans code C qui vérifie la condition de décodage d'ordre *e* (i.e. *e*-correcteur) toutes les boules B(*x*, *e*) / $x \in$C (de rayon *e* centrées en les mots de C) sont disjointes deux à deux et $\bigcup_{x\in \mathrm{C}} \mathrm{B}(x,e) \subset K^n$ et donc $\sum_{x\in \mathrm{C}} card(\mathrm{B}(x,e)) \leq q^n$. Or d'après le Lemme ci-dessus, pour chaque *x* de C, card(B(*x*, *e*)) = $\sum_{i=0}^{i=e} C_n^i\,(q-1)^i$ donc, $\sum_{x\in \mathrm{C}} card(\mathrm{B}(x,e))=\sum_{x\in \mathrm{C}} \sum_{i=0}^{i=e} C_n^i\,(q-1)^i$ et comme $\sum_{i=0}^{i=e} C_n^i\,(q-1)^i$ ne depond pas de *x*, alors $\sum_{x\in \mathrm{C}} card(\mathrm{B}(x,e))= q^k \sum_{i=0}^{i=e} C_n^i\,(q-1)^i$, donc on trouve $q^k \sum_{i=0}^{i=e} C_n^i\,(q-1)^i \leq q^n$ et d'où $q^{n-k} \geq \sum_{i=0}^{i=e} C_n^i\,(q-1)^i$. □

Remarque 3.3.2. Dans le cas d'un code binaire l'inégalité du théorème ci-dessus, sachant que la capacité *e*=$[\frac{d-1}{2}]$, s'écrit comme suit: $\sum_{i=0}^{\left[\frac{d-1}{2}\right]} \binom{n}{i} \leq 2^{n-k}$

La borne de Hamming nous permet de donner une autre définition d'un code parfait.

Définition 3.3.3. Un code linéaire C(n, k, d) sur un corps fini K de cardinal q est dit *code parfait*, si sa borne de Hamming est une égalité, c'est-à-dire si : $\sum_{i=0}^{\left[\frac{d-1}{2}\right]} \binom{n}{i} (q-1)^i = q^{n-k}$.