# TS226
# -
# Codes correcteur d'erreur

**Romain Tajan**

8 octobre 2019

# Plan

1. Codes Linéaires (binaires) en blocs
   - Définition
   - Matrice de parité
   - Encodeur Systématique
   - Détection d'erreur

# Plan

1. Codes Linéaires (binaires) en blocs
   - Définition
   - Matrice de parité
   - Encodeur Systématique
   - Détection d'erreur

# Code linéaire en bloc

## Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$, $\mathcal{C}$ est un **code bianire linéaire** si et seulement si les mots de codes $\mathbf{c} \in \mathbb{F}_2^n$ sont obtenus à partir des messages $\mathbf{u} \in \mathbb{F}_2^k$ par la relation

$$\mathbf{c} = \mathbf{u}G$$

où $G$ est une matrice de taille $k \times n$ appelée **matrice génératrice** de $\mathcal{C}$

$$G = \begin{pmatrix} \mathbf{g_0} \\ \mathbf{g_1} \\ \vdots \\ \mathbf{g_{k-1}} \end{pmatrix} = \begin{pmatrix} g_{0,0} & g_{0,1} & \cdots & g_{0,n-1} \\ g_{1,0} & g_{1,1} & \cdots & g_{1,n-1} \\ \vdots & \vdots & & \vdots \\ g_{k-1,0} & g_{k-1,1} & \cdots & g_{k-1,n-1} \end{pmatrix}$$

## Remarques

1. $\mathcal{C}$ est un sous-espace vectoriel de $\mathbb{F}_2^n$ de dimension $rang(G) = k$
2. Il existe plusieurs matrices génératrices pour un même code.
3. le rendement du code est $R = \frac{rang(G)}{n} = \frac{k}{n}$

# Code dual | Matrice de parité

## Matrice de parité

Le code $\mathcal{C}$ peut aussi être défini par sa **matrice de parité** $H$ de taille $n-k \times n$ :

$$H = \begin{pmatrix} \mathbf{h_0} \\ \mathbf{h_1} \\ \vdots \\ \mathbf{h_{n-k-1}} \end{pmatrix} = \begin{pmatrix} h_{0,0} & h_{0,1} & \cdots & h_{0,n-1} \\ h_{1,0} & h_{1,1} & \cdots & h_{1,n-1} \\ \vdots & \vdots & & \vdots \\ h_{n-k-1,0} & h_{n-k-1,1} & \cdots & h_{n-k-1,n-1} \end{pmatrix}$$

Soit $\mathbf{v} \in \mathbb{F}_2^n$, $\mathbf{v} \in \mathcal{C}$ ($\mathbf{v}$ est un mot de code) si et seulement si

$$\mathbf{v}H^T = 0$$

1. $H$ est appelée **matrice de parité** du code $\mathcal{C}$ et vérifie $GH^T = 0_{k \times n-k}$
2. $H$ n'est pas unique

# Encodeur systématique

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ pour un canal à entrées binaires. Un encodeur $\varphi(\cdot)$ est dit **systématique** ssi

$$\forall \mathbf{u} \in \mathbb{F}_2^k, \varphi(\mathbf{u}) = [\mathbf{p}\ \mathbf{u}] \text{ avec } \mathbf{p} \in \mathbb{F}_2^{n-k}$$

Si $\mathcal{C}$ est linéaire alors il existe une matrice génératrice sous la forme

$$G = \begin{pmatrix} p_{0,0} & \cdots & p_{0,n-k-1} & 1 & 0 & \cdots & 0 \\ p_{1,0} & \cdots & p_{1,n-k-1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{k,0} & \cdots & p_{k,n-k-1} & 0 & 0 & \cdots & 1 \end{pmatrix} = [P \quad I_k]$$

La matrice de parité associée à la matrice $G$ précédente

$$H = \begin{pmatrix} 1 & 0 & \cdots & 0 & p_{0,0} & \cdots & p_{k,0} \\ 0 & 1 & \cdots & 0 & p_{0,1} & \cdots & p_{k,1} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & p_{0,n-k-1} & \cdots & p_{k,n-k-1} \end{pmatrix} = [I_{n-k} \quad P^T]$$

# Remarques sur les encodeurs systématiques

$$G = \begin{pmatrix} p_{0,0} & \cdots & p_{0,n-k-1} & 1 & 0 & \cdots & 0 \\ p_{1,0} & \cdots & p_{1,n-k-1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{k,0} & \cdots & p_{k,n-k-1} & 0 & 0 & \cdots & 1 \end{pmatrix} = [P \quad I_k]$$

1. Un encodeur systématique comporte le message **en clair**

# Remarques sur les encodeurs systématiques

$$G = \begin{pmatrix} p_{0,0} & \cdots & p_{0,n-k-1} & 1 & 0 & \cdots & 0 \\ p_{1,0} & \cdots & p_{1,n-k-1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{k,0} & \cdots & p_{k,n-k-1} & 0 & 0 & \cdots & 1 \end{pmatrix} = [P \quad I_k]$$

1. Un encodeur systématique comporte le message **en clair**
2. Les encodeurs systématiques sont souvent moins complexes que leurs équivalents non-systématiques

# Remarques sur les encodeurs systématiques

$$G = \begin{pmatrix} p_{0,0} & \cdots & p_{0,n-k-1} & 1 & 0 & \cdots & 0 \\ p_{1,0} & \cdots & p_{1,n-k-1} & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ p_{k,0} & \cdots & p_{k,n-k-1} & 0 & 0 & \cdots & 1 \end{pmatrix} = [P \quad I_k]$$

1. Un encodeur systématique comporte le message **en clair**
2. Les encodeurs systématiques sont souvent moins complexes que leurs équivalents non-systématiques
3. Une matrice d'encodage systématique peut être trouvée pour tout code linéaire en bloc de matrice génératrice **pleine** (à des permutations de colonnes près)
   $\rightsquigarrow$ **Pivot de Gauss**

# Exemple de Pivot de Gauss

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

# Exemple de Pivot de Gauss

$$G = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \leftarrow \textbf{Pivot}$$

# Exemple de Pivot de Gauss

$$G = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{matrix} \leftarrow \textbf{Pivot} \\ L_2 \leftarrow L_2 + L_1 \\ \\ \\ \end{matrix}$$

# Exemple de Pivot de Gauss

$$G = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \leftarrow \textbf{Pivot}$$

# Exemple de Pivot de Gauss

$$G = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \begin{matrix} \\ \leftarrow \textbf{Pivot} \\ L_3 \leftarrow L_3 + L_2 \\ \\ \end{matrix}$$

# Exemple de Pivot de Gauss

$$G = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \end{pmatrix} \leftarrow \textbf{Pivot}$$

# Exemple de Pivot de Gauss

$$G = \begin{pmatrix} \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{1} \end{pmatrix} \begin{matrix} \\ \\ \leftarrow \textbf{Pivot} \\ L_4 \leftarrow L_4 + L_3 \end{matrix}$$

# Exemple de Pivot de Gauss

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

1. But : permuter | sommer des lignes pour faire apparaître la matrice $I$ à droite

# Exemple de Pivot de Gauss

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

1. But : permuter | sommer des lignes pour faire apparaître la matrice $I$ à droite
2. Cette procédure ne donne pas tout le temps une matrice de la forme $G = [P, I]$

# Exemple de Pivot de Gauss

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

1. But : permuter | sommer des lignes pour faire apparaître la matrice $I$ à droite
2. Cette procédure ne donne pas tout le temps une matrice de la forme $G = [P, I]$
3. Si $G$ est de rang plein on peut toujours se ramener à $[P, I]$ **à une permutation de colonne près**

# Exemple de Pivot de Gauss

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

1. But : permuter | sommer des lignes pour faire apparaître la matrice $I$ à droite
2. Cette procédure ne donne pas tout le temps une matrice de la forme $G = [P, I]$
3. Si $G$ est de rang plein on peut toujours se ramener à $[P, I]$ **à une permutation de colonne près**
4. Soit $G' = [P, I_k] = G\Pi$ où $\Pi$ est une matrice de permutation des colonnes, soit $H' = [I_{n-k} P^T]$ alors

$$G'(H')^T = 0_{k \times n-k} = GH^T \text{ avec } H = H'\Pi$$

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

Le décodeur peut **détecter une erreur** en calculant le **syndrome**

$$\mathbf{s} = \mathbf{r}H^T$$

Si $\mathbf{s} = \mathbf{0}$ alors $\mathbf{r} \in \mathcal{C}$ sinon il y a une erreur.

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

Le décodeur peut **détecter une erreur** en calculant le **syndrome**

$$\mathbf{s} = \mathbf{r}H^T$$

Si $\mathbf{s} = \mathbf{0}$ alors $\mathbf{r} \in \mathcal{C}$ sinon il y a une erreur.

## Remarques

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

Le décodeur peut **détecter une erreur** en calculant le **syndrome**

$$\mathbf{s} = \mathbf{r}H^T$$

Si $\mathbf{s} = \mathbf{0}$ alors $\mathbf{r} \in \mathcal{C}$ sinon il y a une erreur.

## Remarques

- Les positions des erreurs sont inconnues

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

Le décodeur peut **détecter une erreur** en calculant le **syndrome**

$$\mathbf{s} = \mathbf{r}H^T$$

Si $\mathbf{s} = \mathbf{0}$ alors $\mathbf{r} \in \mathcal{C}$ sinon il y a une erreur.

## Remarques

- Les positions des erreurs sont inconnues
- Certains vecteurs d'erreurs **e** laissent les erreurs non détectées

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

Le décodeur peut **détecter une erreur** en calculant le **syndrome**

$$\mathbf{s} = \mathbf{r}H^T$$

Si $\mathbf{s} = \mathbf{0}$ alors $\mathbf{r} \in \mathcal{C}$ sinon il y a une erreur.

## Remarques

- Les positions des erreurs sont inconnues
- Certains vecteurs d'erreurs $\mathbf{e}$ laissent les erreurs non détectées
- Soit $\mathbf{c}' \in \mathcal{C}$ avec $\mathbf{c}' \neq \mathbf{c}$, il suffit de prendre $\mathbf{e} = \mathbf{c} + \mathbf{c}'$

# Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$.

Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

Le décodeur peut **détecter une erreur** en calculant le **syndrome**

$$\mathbf{s} = \mathbf{r}H^T$$

Si $\mathbf{s} = \mathbf{0}$ alors $\mathbf{r} \in \mathcal{C}$ sinon il y a une erreur.

## Remarques

- Les positions des erreurs sont inconnues
- Certains vecteurs d'erreurs $\mathbf{e}$ laissent les erreurs non détectées
- Soit $\mathbf{c}' \in \mathcal{C}$ avec $\mathbf{c}' \neq \mathbf{c}$, il suffit de prendre $\mathbf{e} = \mathbf{c} + \mathbf{c}'$
- Dans ce cas $\mathbf{r} = \mathbf{c}'$ et comme $\mathbf{c}' \in \mathcal{C}$, $\mathbf{r}H^T = \mathbf{0}$

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (} \mathbf{e} \text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

## Remarques

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (} \mathbf{e} \text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

## Remarques

- **Poids de Hamming** : soit $\mathbf{v} = [v_0, v_1, \ldots v_{n-1}] \in \mathbb{F}_2^n$ alors $w_H(\mathbf{v}) = |\{i : v_i = 1\}|$

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

## Remarques

- **Poids de Hamming** : soit $\mathbf{v} = [v_0, v_1, \ldots v_{n-1}] \in \mathbb{F}_2^n$ alors $w_H(\mathbf{v}) = |\{i : v_i = 1\}|$
- **Distance de Hamming** : soient $\mathbf{v}, \mathbf{v}' \in \mathbb{F}_2^n$ alors $d_H(\mathbf{v}, \mathbf{v}') = \left|\left\{i : v_i \neq v_i'\right\}\right|$

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (}\mathbf{e}\text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

## Remarques

- **Poids de Hamming** : soit $\mathbf{v} = [v_0, v_1, \ldots v_{n-1}] \in \mathbb{F}_2^n$ alors $w_H(\mathbf{v}) = |\{i : v_i = 1\}|$
- **Distance de Hamming** : soient $\mathbf{v}, \mathbf{v}' \in \mathbb{F}_2^n$ alors $d_H(\mathbf{v}, \mathbf{v}') = \left|\left\{i : v_i \neq v_i'\right\}\right|$
- La séquence $A_i$ est appelée **spectre de poids** de $\mathcal{C}$

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (} \mathbf{e} \text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

## Remarques

- **Poids de Hamming** : soit $\mathbf{v} = [v_0, v_1, \ldots v_{n-1}] \in \mathbb{F}_2^n$ alors $w_H(\mathbf{v}) = |\{i : v_i = 1\}|$
- **Distance de Hamming** : soient $\mathbf{v}, \mathbf{v}' \in \mathbb{F}_2^n$ alors $d_H(\mathbf{v}, \mathbf{v}') = \left|\left\{i : v_i \neq v_i'\right\}\right|$
- La séquence $A_i$ est appelée **spectre de poids** de $\mathcal{C}$
- La plus petite valeur de $i$ telle que $A_i \neq 0$ est appelée **distance minimale** de $\mathcal{C}$

# Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $(2^k, n)$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

$$\mathbf{r} = \mathbf{c} + \mathbf{e} \text{ (} \mathbf{e} \text{ est apelé vecteur d'erreur )}$$

On cherche ici la **probabilité d'une erreur non détectée**

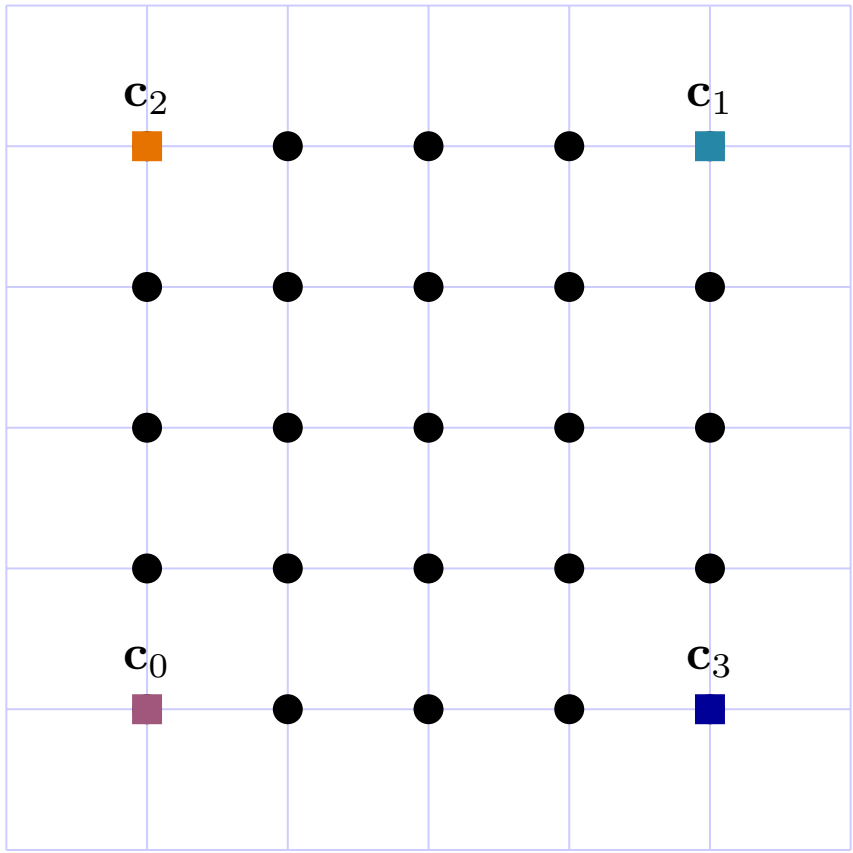
$$P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$$

où $A_i$ est le nombre de mots de codes non-nuls de $\mathcal{C}$ de poids de Hamming $w_H(\mathbf{c}) = i$

## Remarques

- **Poids de Hamming** : soit $\mathbf{v} = [v_0, v_1, \ldots v_{n-1}] \in \mathbb{F}_2^n$ alors $w_H(\mathbf{v}) = |\{i : v_i = 1\}|$
- **Distance de Hamming** : soient $\mathbf{v}, \mathbf{v}' \in \mathbb{F}_2^n$ alors $d_H(\mathbf{v}, \mathbf{v}') = \left|\left\{i : v_i \neq v_i'\right\}\right|$
- La séquence $A_i$ est appelée **spectre de poids** de $\mathcal{C}$
- La plus petite valeur de $i$ telle que $A_i \neq 0$ est appelée **distance minimale** de $\mathcal{C}$
- Un code $\mathcal{C}$ de distance minimale $d_{min}(\mathcal{C})$ peut **détecter** toute erreur de poids inférieur à $d_{min} - 1$

# Démonstration de $P_U(E) = \sum_i A_i p^i (1-p)^{n-i}$




## Remarques

- **Poids de Hamming** : soit $\mathbf{v} = [v_0, v_1, \dots v_{n-1}] \in \mathbb{F}_2^n$ alors $w_H(\mathbf{v}) = |\{i : v_i = 1\}|$
- **Distance de Hamming** : soient $\mathbf{v}, \mathbf{v}' \in \mathbb{F}_2^n$ alors $d_H(\mathbf{v}, \mathbf{v}') = \left|\{i : v_i \neq v'_i\}\right|$

# Spectre des poids $\{A_i\}_{i \geq d_{min}}$

# Correction d'erreurs

Mettre la question 5