TS226
-
Codes correcteurs d’erreurs

Romain Tajan

8 septembre 2025

Plan

Sur les codes en blocs binaires

Définitions

1. Un code est dit **binaire** si $\mathcal{X} = \mathbb{F}_2 = \{0, 1\}$.

Sur les codes en blocs binaires

Définitions

1. Un code est dit **binaire** si $\mathcal{X} = \mathbb{F}_2 = \{0, 1\}$.
2. Si les messages w représentent des séquences binaires de taille k alors le rendement

$$R = \frac{k}{n}$$

Sur les codes en blocs binaires

Définitions

1. Un code est dit **binaire** si $\mathcal{X} = \mathbb{F}_2 = \{0, 1\}$.
2. Si les messages w représentent des séquences binaires de taille k alors le rendement

$$R = \frac{k}{n}$$

3. On appelle **distance minimale du code** $\mathcal{C}$ la quantité suivante :

$$d_{min}(\mathcal{C}) = \min_{\mathbf{c}\in\mathcal{C},\mathbf{c}'\in\mathcal{C},\mathbf{c}'\neq\mathbf{c}} d_H(\mathbf{c}, \mathbf{c}')$$

Sur les codes en blocs binaires

Définitions

1. Un code est dit **binaire** si $\mathcal{X} = \mathbb{F}_2 = \{0, 1\}$.
2. Si les messages w représentent des séquences binaires de taille k alors le rendement

$$R = \frac{k}{n}$$

3. On appelle **distance minimale du code** $\mathcal{C}$ la quantité suivante :

$$d_{min}(\mathcal{C}) = \min_{\mathbf{c} \in \mathcal{C}, \mathbf{c}' \in \mathcal{C}, \mathbf{c}' \neq \mathbf{c}} d_H(\mathbf{c}, \mathbf{c}')$$

4. On notera $[n, k, d]$ un code $(2^k, n)$ binaire de distance minimale d

It’s quizz time !

Préparez vos téléphones !

QCM

Considérons l'encodage de 1 bit par un code à 3 répétitions. Son rendement vaut

- A $1/2$
- B $1/3$
- C $1/4$
- D $1/5$

#QDLE#Q#AB*CD#30#

QCM

Considérons l'encodage de 1 bit par un code à 3 répétitions. La distance minimale du code engendré est :

- A 1
- B 2
- C 3
- D 4

#QDLE#Q#ABC*D#45#

Propriétés des codes $[n, k, d]$

Propriétés

Tout code $[n, k, d]$ vérifie les propriétés suivantes :

1. **Borne de Singleton** : $d \leq n - k + 1$.
   Un code tel que $d = n - k + 1$ est dit **Maximum Distance Separable** ou **MDS**

Propriétés des codes $[n, k, d]$

Propriétés

Tout code $[n, k, d]$ vérifie les propriétés suivantes :

1. **Borne de Singleton** : $d \leq n - k + 1$.
   Un code tel que $d = n - k + 1$ est dit **Maximum Distance Separable** ou **MDS**
2. Sur canal **BSC**, toutes les combinaisons de $d - 1$ **erreurs** (ou moins) peuvent être **détectées**.

Propriétés des codes $[n, k, d]$

Propriétés

Tout code $[n, k, d]$ vérifie les propriétés suivantes :

1. **Borne de Singleton** : $d \leq n - k + 1$.
   Un code tel que $d = n - k + 1$ est dit **Maximum Distance Separable** ou **MDS**
2. Sur canal **BSC**, toutes les combinaisons de $d - 1$ **erreurs** (ou moins) peuvent être **détectées**.
3. Sur canal **BSC**, toutes les combinaisons de $\lfloor (d-1)/2 \rfloor$ **erreurs** (ou moins) peuvent être **corrigées**.

It's quizz time !

Code de parité simple

Le code de parité simple pour des messages de k bits est le code de taille $n = k + 1$ tel que :

1. les k premiers bits sont une recopie des bits de message
2. le bit $k + 1$ est :

$$\begin{cases} 1, \text{ si le message comporte un nombre pair de bits à } 1 \\ 0, \text{ sinon} \end{cases}$$

Préparez vos téléphones !

QCM

Combien vaut La distance minimale du code de parité ?

- A 1
- B 2
- C 3
- D 4

#QDLE#Q#AB*CD#60#

QCM

Le code de parité est un code MDS.

- (A) Vrai
- (B) Faux

#QDLE#Q#A*B#60#

QCM

Combien d'erreurs sur le canal BSC le code de parité simple de taille $n = 8$ bits peut-il détecter ?

- A 8
- B 7
- C 4
- D 1

#QDLE#Q#ABCD*#60#

QCM

Le code de parité simple de taille $n = k + 1$ ne peut pas corriger d'erreur.

- A Vrai
- B Faux
- C Je ne sais pas.

#QDLE#Q#A*BCD#60#

Plan

Avant de commencer...

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**

Avant de commencer...

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
    - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)

Avant de commencer...

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
   - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)
   - Pour $x, y \in \mathbb{F}_2$, $x \cdot y$ est le produit "classique" entre x et y ($\equiv$ ET)

Avant de commencer...

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
   - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)
   - Pour $x, y \in \mathbb{F}_2$, $x \cdot y$ est le produit "classique" entre x et y ($\equiv$ ET)
3. $\mathbb{F}_2$ est un corps fini à deux éléments ($\mathbb{Z}/2\mathbb{Z}$)

Avant de commencer...

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
   - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)
   - Pour $x, y \in \mathbb{F}_2$, $x \cdot y$ est le produit "classique" entre x et y ($\equiv$ ET)
3. $\mathbb{F}_2$ est un corps fini à deux éléments ($\mathbb{Z}/2\mathbb{Z}$)
4. Par la suite on notera $\oplus \rightsquigarrow +$

Avant de commencer...

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
   - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)
   - Pour $x, y \in \mathbb{F}_2$, $x \cdot y$ est le produit "classique" entre x et y ($\equiv$ ET)
3. $\mathbb{F}_2$ est un corps fini à deux éléments ($\mathbb{Z}/2\mathbb{Z}$)
4. Par la suite on notera $\oplus \rightsquigarrow +$
5. $(\mathbb{F}_2^n, +, \cdot)$ est un **espace vectoriel** où
   - Pour $\mathbf{x}, \mathbf{y} \in \mathbb{F}_2^n$, $\quad \mathbf{x} + \mathbf{y} = [x_0 + y_0, x_1 + y_1 \ldots, x_{n-1} + y_{n-1}]$
   - Pour $x \in \mathbb{F}_2$ et $\mathbf{y} \in \mathbb{F}_2^n$, $\quad x \cdot \mathbf{y} = [x \cdot y_0, x \cdot y_1, \ldots, x \cdot y_{n-1}]$

It’s quizz time !

Préparez vos téléphones !

Je suis à l'aise avec les notions d'algèbre telles que les notions de **groupe** / **anneau** / **corps** et **espace vectoriel**.

- A Après un peu de révision oui.
- B Après beaucoup de révision oui.
- C C'est très compliqué.
- D Pas du tout.

#QDLE#S#ABCD#20#

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
   - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)
   - Pour $x, y \in \mathbb{F}_2$, $x \cdot y$ est le produit "classique" entre x et y ($\equiv$ ET)

Dans $\mathbb{F}_2$ que vaut $x + x$?

- A 0
- B 1
- C x
- D $\bar{x}$

#QDLE#Q#A*BCD#30#

Remarques

1. Dans cette section $\mathcal{X} = \mathcal{Y} = \{0, 1\}$ et le canal considéré est le **canal binaire symétrique**
2. Dans cette section on notera $\mathbb{F}_2$ le **corps** $(\{0, 1\}, \oplus, \cdot)$ où :
   - Pour $x, y \in \mathbb{F}_2$, $x \oplus y = (x + y) \mod 2$ ($\equiv$ OU exclusif)
   - Pour $x, y \in \mathbb{F}_2$, $x \cdot y$ est le produit "classique" entre x et y ($\equiv$ ET)

Dans $\mathbb{F}_2$ que vaut $x \cdot x$?

- A. 0
- B. 1
- C. x
- D. $\bar{x}$

#QDLE#Q#ABC*D#30#

Dire si l'assertion suivante est vraie. "Si $\mathbf{x} \in \mathbb{F}_2^n$, alors $-\mathbf{x} = \mathbf{x}$."

- A Vrai
- B Faux

#QDLE#Q#A*B#30#

Code linéaire en bloc

Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$.
$\mathcal{C}$ est dit **linéaire** si et seulement si, il existe k vecteurs $\mathbf{g}_0, \mathbf{g}_1, \ldots, \mathbf{g}_{k-1} \in \mathbb{F}_2^n$ tels que, pour tout $\mathbf{c} \in \mathcal{C}$,

$$\mathbf{c} = \sum_{i=0}^{k-1} u_i \mathbf{g}_i$$

avec $u_i \in \mathbb{F}_2$

Remarques

1. L'ensemble $\mathcal{B}_{\mathcal{C}} = \{\mathbf{g}_0, \mathbf{g}_1, \ldots, \mathbf{g}_{k-1}\}$ est appelé **base** de $\mathcal{C}$.
2. $\mathcal{C}$ est un sous-espace vectoriel de $\mathbb{F}_2^n$ de dimension k (si $\mathcal{B}_{\mathcal{C}}$ est une base libre)

Matrice Génératrice

Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, il existe une matrice G de taille $k \times n$ telle que pour tout $\mathbf{c} \in \mathcal{C}$,

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

Par définition on a

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

1. G est appelé **matrice génératrice** du code $\mathcal{C}$

Matrice Génératrice

Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, il existe une matrice G de taille $k \times n$ telle que pour tout $\mathbf{c} \in \mathcal{C}$,

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

Par définition on a

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

1. G est appelé **matrice génératrice** du code $\mathcal{C}$
2. Pour ce cours G est de **rang plein**

Matrice Génératrice

Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, il existe une matrice G de taille $k \times n$ telle que pour tout $\mathbf{c} \in \mathcal{C}$,

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

Par définition on a

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

1. G est appelé **matrice génératrice** du code $\mathcal{C}$
2. Pour ce cours G est de **rang plein**
3. Pour un code $\mathcal{C}$, il existe plusieurs matrices génératrices

Matrice Génératrice

Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, il existe une matrice G de taille $k \times n$ telle que pour tout $\mathbf{c} \in \mathcal{C}$,

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

Par définition on a

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

1. G est appelé **matrice génératrice** du code $\mathcal{C}$
2. Pour ce cours G est de **rang plein**
3. Pour un code $\mathcal{C}$, il existe plusieurs matrices génératrices
4. Permuter / combiner les lignes de G ne change pas $\mathcal{C}$

Matrice Génératrice

Code linéaire

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, il existe une matrice G de taille $k \times n$ telle que pour tout $\mathbf{c} \in \mathcal{C}$,

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

Par définition on a

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

1. G est appelé **matrice génératrice** du code $\mathcal{C}$
2. Pour ce cours G est de **rang plein**
3. Pour un code $\mathcal{C}$, il existe plusieurs matrices génératrices
4. Permuter / combiner les lignes de G ne change pas $\mathcal{C}$
5. Permuter les colonnes de G change l'espace $\mathcal{C}$ mais ne change pas les performances du code

It’s quizz time !

Préparez vos téléphones !

Parmis les matrices proposées, lequelles sont des matrices génératrices pour le code de parité [5, 4, 2]

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

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

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

- A A seulement
- B B seulement
- C C seulement
- D A et B seulement
- E B et C seulement
- F A et C seulement
- G A, B et C
- H Aucune

#QDLE#Q#ABCDEFG*H#60#

Code dual | Matrice de parité

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, on appelle **code dual** :

$$\mathcal{C}_d = \left\{\mathbf{v} \in \mathbb{F}_2^n : \forall \mathbf{c} \in \mathcal{C} \;\; < \mathbf{v}, \mathbf{c} >= 0\right\}$$

où $< \mathbf{v}, \mathbf{c} >= \sum_{i=0}^{n-1} v_i c_i$

1. La dimension du sous-espace vectoriel $\mathcal{C}_d$ est $n - k$

Code dual | Matrice de parité

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, on appelle **code dual** :

$$\mathcal{C}_d = \left\{\mathbf{v} \in \mathbb{F}_2^n : \forall \mathbf{c} \in \mathcal{C} \ < \mathbf{v}, \mathbf{c} >= 0\right\}$$

où $< \mathbf{v}, \mathbf{c} >= \sum_{i=0}^{n-1} v_i c_i$

1. La dimension du sous-espace vectoriel $\mathcal{C}_d$ est $n - k$
2. Soit $\mathcal{B}_{\mathcal{C}_d} = \{\mathbf{h}_0, \mathbf{h}_1, \dots, \mathbf{h}_{n-k-1}\}$ une base de $\mathcal{C}_d$, alors $\mathcal{C}_d$ a pour matrice génératrice

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

Code dual | Matrice de parité

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, on appelle **code dual** :

$$\mathcal{C}_d = \left\{\mathbf{v} \in \mathbb{F}_2^n : \forall \mathbf{c} \in \mathcal{C} \ \ < \mathbf{v}, \mathbf{c} >= 0\right\}$$

où $< \mathbf{v}, \mathbf{c} >= \sum_{i=0}^{n-1} v_i c_i$

1. La dimension du sous-espace vectoriel $\mathcal{C}_d$ est $n - k$
2. Soit $\mathcal{B}_{\mathcal{C}_d} = \{\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_{n-k-1}\}$ une base de $\mathcal{C}_d$, alors $\mathcal{C}_d$ a pour matrice génératrice

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

3. Le code $\mathcal{C}$ peut être défini comme $\mathcal{C} = \left\{\mathbf{c} \in \mathbb{F}_2^n : \mathbf{c}H^T = \mathbf{0}\right\} (= (\mathcal{C}_d)_d)$

Code dual | Matrice de parité

Soit $\mathcal{C}$ un code $(M = 2^k, n)$ linéaire, on appelle **code dual** :

$$\mathcal{C}_d = \left\{\mathbf{v} \in \mathbb{F}_2^n : \forall \mathbf{c} \in \mathcal{C} \;\; < \mathbf{v}, \mathbf{c} >= 0\right\}$$

où $< \mathbf{v}, \mathbf{c} >= \sum_{i=0}^{n-1} v_i c_i$

1. La dimension du sous-espace vectoriel $\mathcal{C}_d$ est $n - k$
2. Soit $\mathcal{B}_{\mathcal{C}_d} = \{\mathbf{h}_0, \mathbf{h}_1, \ldots, \mathbf{h}_{n-k-1}\}$ une base de $\mathcal{C}_d$, alors $\mathcal{C}_d$ a pour matrice génératrice

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

3. Le code $\mathcal{C}$ peut être défini comme $\mathcal{C} = \left\{\mathbf{c} \in \mathbb{F}_2^n : \mathbf{c}H^T = \mathbf{0}\right\} (= (\mathcal{C}_d)_d)$
4. H est appelée matrice de parité du code $\mathcal{C}$ et vérifie $GH^T = 0_{k \times n-k}$

Encodeur systématique

Soit $\mathcal{C}$ un code linéaire $[n, k, d]$ 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

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

La matrice de parité associée à la matrice G précédente

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

Remarques sur les encodeurs systématiques

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

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

Remarques sur les encodeurs systématiques

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

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

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

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

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

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 Π est une matrice de permutation des colonnes, soit $H' = [I_{n-k} P^T]$ alors

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

Exemple de Pivot de Gauss

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

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 Π est une matrice de permutation des colonnes, soit $H' = [I_{n-k}P^T]$ alors

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

Exemple de Pivot de Gauss

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

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 Π est une matrice de permutation des colonnes, soit $H' = [I_{n-k} P^T]$ alors

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

Exemple de Pivot de Gauss

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

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 Π est une matrice de permutation des colonnes, soit $H' = [I_{n-k} P^T]$ alors

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

Exemple de Pivot de Gauss

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

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 Π est une matrice de permutation des colonnes, soit $H' = [I_{n-k} P^T]$ alors

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

Exemple de Pivot de Gauss

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

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 Π est une matrice de permutation des colonnes, soit $H' = [I_{n-k} P^T]$ alors

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

Exemple de Pivot de Gauss

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

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 Π 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 $[n, k, d]$.

Détection d'erreurs dans le canal BSC

Soit $\mathcal{C}$ un code linéaire en bloc $[n, k, d]$.

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 $[n, k, d]$.

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 $[n, k, d]$.

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 $[n, k, d]$.

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 $[n, k, d]$.

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 $[n, k, d]$.

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 $[n, k, d]$.

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 $[n, k, d]$. 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 $[n, k, d]$. 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 $[n, k, d]$. 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 $[n, k, d]$. 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 $d_H(\mathbf{v}, \mathbf{v}') = \left|\{i : v_i \neq v'_i\}\right|$

Probabilité d'une erreur non détectée

Soit $\mathcal{C}$ un code linéaire en bloc $[n, k, d]$. 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 $d_H(\mathbf{v}, \mathbf{v}') = \left|\{i : v_i \neq v_i'\}\right|$
- 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 $[n, k, d]$. 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 $d_H(\mathbf{v}, \mathbf{v}') = \left|\{i : v_i \neq v_i'\}\right|$
- 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 $[n, k, d]$. Soit $\mathbf{c} \in \mathcal{C}$ le mot de code transmis et soit $\mathbf{r}$ le mot reçu

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

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 $d_H(\mathbf{v}, \mathbf{v}') = \left|\{i : v_i \neq v_i'\}\right|$
- 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 peut **détecter** toute erreur de poids inférieur à $d - 1$

It's quizz time !

Préparez vos téléphones !

Soit $\mathcal{C}$ le code de Hamming ayant pour matrice de parité

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

Parmis tous les candidats, trouver le mot de code de $\mathcal{C}$

- A $\mathbf{c} = [1\ 1\ 1\ 1\ 1\ 1\ 0]$
- B $\mathbf{c} = [0\ 1\ 0\ 1\ 0\ 1\ 0]$
- C $\mathbf{c} = [0\ 1\ 0\ 1\ 1\ 1\ 1]$
- D $\mathbf{c} = [1\ 1\ 0\ 1\ 1\ 1\ 0]$

#QDLE#Q#AB*CD#60#

Soit $\mathcal{C}$ le code de Hamming ayant pour matrice de parité

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

On reçoit le mot $\mathbf{r} = [0\ 1\ 1\ 1\ 1\ 1\ 0]$, trouver l'assertion vraie

- A Le syndrome est $= [1\ 1\ 1]$, la transmission est erronée.
- B Le syndrome est $= [1\ 1\ 0]$, la transmission est erronée.
- C Le syndrome est $= [0\ 1\ 1]$, la transmission est erronée.
- D La transmission s'est passée sans erreur.

#QDLE#Q#ABC*D#60#

Soit $\mathcal{C}$ le code de Hamming ayant pour matrice de parité

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

Trouver l'assertion vraie

- A La distance minimale de ce code est au plus de 3.
- B Ce code peut corriger 2 erreurs.
- C Ce code a une longueur 4 bits.
- D Ce code permet d'encoder des messages de 3 bits.

#QDLE#Q#A*BCD#60#

Décodage par syndrome

- Il y a 2^{n-k} syndromes différents

Décodage par syndrome

- Il y a 2^{n-k} syndromes différents
- Il y a 2^k mots différents.

Décodage par syndrome

- Il y a 2^{n-k} syndromes différents
- Il y a 2^k mots différents.
- On construit un tableau de la manière suivante :
    1. Les colonnes représentent les mots de codes possibles
    2. Les lignes représentent les "vecteurs d'erreurs" possibles
    3. La première ligne est obtenue en considérant le vecteur d'erreur $\mathbf{0}$
    4. Supposons les $j-1$ premières lignes construites, e_j est choisi parmi les éléments de $\mathcal{C}^{\perp}$ n'étant pas déjà dans le tableau
    5. La ligne j, est $e_j + \mathcal{C} = \{e_j + \mathbf{c} : \mathbf{c} \in \mathcal{C}\}$

Décodage par syndrome

- Il y a 2^{n-k} syndromes différents
- Il y a 2^k mots différents.
- On construit un tableau de la manière suivante :
  1. Les colonnes représentent les mots de codes possibles
  2. Les lignes représentent les "vecteurs d'erreurs" possibles
  3. La première ligne est obtenue en considérant le vecteur d'erreur **0**
  4. Supposons les $j-1$ premières lignes construites, e_j est choisi parmi les éléments de $\mathcal{C}^\perp$ n'étant pas déjà dans le tableau
  5. La ligne j, est $e_j + \mathcal{C} = \{e_j + \mathbf{c} : \mathbf{c} \in \mathcal{C}\}$

	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\dots$	$\mathbf{c}_{2^k-1}$
$\mathbf{0}$	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\dots$	$\mathbf{c}_{2^k-1}$

Décodage par syndrome

- Il y a 2^{n-k} syndromes différents
- Il y a 2^k mots différents.
- On construit un tableau de la manière suivante :
  1. Les colonnes représentent les mots de codes possibles
  2. Les lignes représentent les "vecteurs d'erreurs" possibles
  3. La première ligne est obtenue en considérant le vecteur d'erreur **0**
  4. Supposons les $j-1$ premières lignes construites, e_j est choisi parmi les éléments de $\mathcal{C}^\perp$ n'étant pas déjà dans le tableau
  5. La ligne j, est $e_j + \mathcal{C} = \{e_j + \mathbf{c} : \mathbf{c} \in \mathcal{C}\}$

	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\dots$	$\mathbf{c}_{2^k-1}$
$\mathbf{0}$	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\dots$	$\mathbf{c}_{2^k-1}$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\mathbf{e}_1+\mathbf{c}_1$	$\mathbf{e}_1+\mathbf{c}_2$	$\dots$	$\mathbf{e}_1+\mathbf{c}_{2^k-1}$

Décodage par syndrome

- Il y a 2^{n-k} syndromes différents
- Il y a 2^k mots différents.
- On construit un tableau de la manière suivante :
  1. Les colonnes représentent les mots de codes possibles
  2. Les lignes représentent les "vecteurs d'erreurs" possibles
  3. La première ligne est obtenue en considérant le vecteur d'erreur **0**
  4. Supposons les $j-1$ premières lignes construites, e_j est choisi parmi les éléments de $\mathcal{C}^{\perp}$ n'étant pas déjà dans le tableau
  5. La ligne j, est $e_j + \mathcal{C} = \{e_j + \mathbf{c} : \mathbf{c} \in \mathcal{C}\}$

	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\ldots$	$\mathbf{c}_{2^k-1}$
$\mathbf{0}$	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\ldots$	$\mathbf{c}_{2^k-1}$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\mathbf{e}_1+\mathbf{c}_1$	$\mathbf{e}_1+\mathbf{c}_2$	$\ldots$	$\mathbf{e}_1+\mathbf{c}_{2^k-1}$
$\mathbf{e}_2$	$\mathbf{e}_2$	$\mathbf{e}_2+\mathbf{c}_1$	$\mathbf{e}_2+\mathbf{c}_2$	$\ldots$	$\mathbf{e}_2+\mathbf{c}_{2^k-1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\mathbf{e}_{2^{n-k}-1}$	$\mathbf{e}_{2^{n-k}-1}$	$\mathbf{e}_{2^{n-k}-1}+\mathbf{c}_1$	$\mathbf{e}_{2^{n-k}-1}+\mathbf{c}_2$	$\ldots$	$\mathbf{e}_{2^{n-k}-1}+\mathbf{c}_{2^k-1}$

Décodage par syndrome

	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\dots$	$\mathbf{c}_{2^k-1}$
$\mathbf{0}$	$\mathbf{0}$	$\mathbf{c}_1$	$\mathbf{c}_2$	$\dots$	$\mathbf{c}_{2^k-1}$
$\mathbf{e}_1$	$\mathbf{e}_1$	$\mathbf{e}_1+\mathbf{c}_1$	$\mathbf{e}_1+\mathbf{c}_2$	$\dots$	$\mathbf{e}_1+\mathbf{c}_{2^k-1}$
$\mathbf{e}_2$	$\mathbf{e}_2$	$\mathbf{e}_2+\mathbf{c}_1$	$\mathbf{e}_2+\mathbf{c}_2$	$\dots$	$\mathbf{e}_2+\mathbf{c}_{2^k-1}$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		$\vdots$
$\mathbf{e}_{2^{n-k}-1}$	$\mathbf{e}_{2^{n-k}-1}$	$\mathbf{e}_{2^{n-k}-1}+\mathbf{c}_1$	$\mathbf{e}_{2^{n-k}-1}+\mathbf{c}_2$	$\dots$	$\mathbf{e}_{2^{n-k}-1}+\mathbf{c}_{2^k-1}$

Propriétés

1. Toutes les lignes du tableau (appelées **coset**) sont différentes.
2. Toutes les colonnes du tableau sont différentes.
3. **Tous les éléments d'une même ligne ont le même syndrome !**

Décodage par syndrome

Décodage

1. On considère e_j comme étant un élément de poids minimum sur la ligne j
2. Calculer le syndrome : $= \mathbf{r}H^T$
3. Trouver j tel que $= \mathbf{e}_j H^T$
4. Décoder $\hat{c} = \mathbf{r} + \mathbf{e}$

$$
\begin{array}{c|cccccc}
 & \mathbf{0} & \mathbf{c}_1 & \mathbf{c}_2 & \cdots & \mathbf{c}_{2^k-1} \\
\hline
\mathbf{0} & \mathbf{0} & \mathbf{c}_1 & \mathbf{c}_2 & \cdots & \mathbf{c}_{2^k-1} \\
\mathbf{e}_1 & \mathbf{e}_1 & \mathbf{e}_1 + \mathbf{c}_1 & \mathbf{e}_1 + \mathbf{c}_2 & \cdots & \mathbf{e}_1 + \mathbf{c}_{2^k-1} \\
\mathbf{e}_2 & \mathbf{e}_2 & \mathbf{e}_2 + \mathbf{c}_1 & \mathbf{e}_2 + \mathbf{c}_2 & \cdots & \mathbf{e}_2 + \mathbf{c}_{2^k-1} \\
\vdots & \vdots & \vdots & \vdots & & \vdots \\
\mathbf{e}_{2^{n-k}-1} & \mathbf{e}_{2^{n-k}-1} & \mathbf{e}_{2^{n-k}-1} + \mathbf{c}_1 & \mathbf{e}_{2^{n-k}-1} + \mathbf{c}_2 & \cdots & \mathbf{e}_{2^{n-k}-1} + \mathbf{c}_{2^k-1}
\end{array}
$$

Dernier QCM

Comment avez-vous trouvé ce cours ?

- A Très difficile
- B Difficile
- C Moyen
- D Simple
- E Très simple

#QDLE#S#ABCDE#30#

Démonstration de la borne de Singleton

Soit $\mathcal{C}$ un code binaire $[n, k, d]$ ce code possède 2^k mots de codes **différents** parmi les 2^n mots possibles de taille n.

Pour chaque mot de code dans $\mathcal{C}$, retirons $d - 1$ composantes. Les vecteurs ainsi obtenus sont encore tous différents. En effet, deux mots de code différents diffèrent par au moins d valeurs (cf définition de la distance minimale).

Le nouveau code ainsi construit possède donc 2^k mots de codes différents de taille $n - d + 1$. Or il y a 2^{n-d+1} mots de taille $n - d + 1$. D'où on a $2^{n-d+1} \geq 2^k$, ce qui fait :

$$d \leq n - k + 1 \quad \textbf{borne de Singleton}$$

Notes que si le code avait été non-binaire (ternaire, quaternaire...), le résultat resterait vrai.

Démonstration du nombre d'erreurs détectables

Soit $\mathcal{C}$ un code binaire $[n, k, d]$ considéré sur canal BSC.

Soient $\mathbf{x} \in \mathcal{C} \subseteq \mathbb{F}_2^n$ et $\mathbf{y} \in \mathbb{F}_2^n$ représentant respectivement le mot de code transmis le vecteur observé.

Si $d_H(\mathbf{y}, \mathbf{x}) \leq d - 1$ alors $\mathbf{y}$ ne peut être un mot du code. En effet, la distance minimale de $\mathcal{C}$ étant d, deux mots différents dans $\mathcal{C}$ diffèrent sur au moins d éléments. Donc l'erreur est détectée en vérifiant que $\mathbf{y} \notin \mathcal{C}$.

Si on avait d éléments, alors il serait possible de trouver $\mathbf{x}$ et un schéma d'erreur tels que $\mathbf{y} \in \mathcal{C}$, rendant ce schéma d'erreur indétectable.

On a donc démontré que tout schéma d'au plus $d - 1$ erreurs peut être détecté.

Démonstration du nombre d'erreurs corrigibles

Soit $\mathcal{C}$ un code binaire $[n, k, d]$ considéré sur canal BSC.

On va procéder par l'absurde. Supposons que le canal ait introduit un nombre d'erreur inférieur à $(d-1)/2$, i.e. $d_H(\mathbf{x}_1, \mathbf{y}) \leq (d-1)/2$ et que le décodage du MV ait échoué : $\mathbf{x}_2$ est décidé au lieu de $\mathbf{x}_1$ qui a été envoyé(avec $\mathbf{x}_1 \neq \mathbf{x}_2$).

Sur canal BSC, le décodage MV revient à chercher le mot de code le plus proche de $\mathbf{y}$ au sens de la distance de Hamming (nombre de différences). Comme $\mathbf{x}_2$ est décidé à la place de $\mathbf{x}_1$ on a

$$d_H(\mathbf{y}, \mathbf{x}_2) \leq d_H(\mathbf{y}, \mathbf{x}_1) \leq (d-1)/2$$

d'où

$$d_H(\mathbf{y}, \mathbf{x}_2) + d_H(\mathbf{y}, \mathbf{x}_1) \leq d - 1 < d$$

Or, $\mathcal{C}$ ayant une distance minimale d on a que $d_H(\mathbf{x}_2, \mathbf{x}_1) \geq d$. Enfin l'inégalité triangulaire pour la distance d_H donne

$$d_H(\mathbf{y}, \mathbf{x}_1) + d_H(\mathbf{y}, \mathbf{x}_2) \geq d_H(\mathbf{x}_2, \mathbf{x}_1) \geq d$$

Ce qui est contradictoire avec l'inégalité démontrée plus haut.

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

$$
\begin{aligned}
P_U(E) &= \mathbb{P}(\mathbf{R} \in \mathcal{C}) \\
&= \sum_{\mathbf{x} \in \mathcal{C}} \mathbb{P}(\mathbf{R} \in \mathcal{C} | \mathbf{X} = \mathbf{x}) \mathbb{P}(\mathbf{X} = \mathbf{x}) \\
&= \sum_{\mathbf{x} \in \mathcal{C}} \mathbb{P}(\Cup_{\mathbf{r} \in \mathcal{C} - \{\mathbf{x}\}} \mathbf{R} = \mathbf{r} | \mathbf{X} = \mathbf{x}) \mathbb{P}(\mathbf{X} = \mathbf{x}) \\
&= \frac{1}{M} \sum_{\mathbf{x} \in \mathcal{C}} \sum_{\mathbf{r} \in \mathcal{C} - \{\mathbf{x}\}} \mathbb{P}(\mathbf{R} = \mathbf{r} | \mathbf{X} = \mathbf{x})
\end{aligned}
$$

Or, comme on a vu que $\mathbb{P}(\mathbf{R} = \mathbf{r} | \mathbf{X} = \mathbf{x}) = p^{d_H(\mathbf{x},\mathbf{r})}(1-p)^{n-d_H(\mathbf{x},\mathbf{r})}$ et que $d_H(\mathbf{x}, \mathbf{r}) = w_H(\mathbf{x} + \mathbf{r}) = d_H(\mathbf{0}, \mathbf{r} + \mathbf{x})$ on a

$$
\begin{aligned}
P_U(E) &= \frac{1}{M} \sum_{\mathbf{x} \in \mathcal{C}} \sum_{\mathbf{r} \in \mathcal{C} - \{\mathbf{x}\}} p^{d_H(\mathbf{0},\mathbf{x}+\mathbf{r})}(1-p)^{n-d_H(\mathbf{0},\mathbf{x}+\mathbf{r})} \\
&= \frac{1}{M} \sum_{\mathbf{x} \in \mathcal{C}} \sum_{\mathbf{x}' \in \mathcal{C} - \{\mathbf{0}\}} p^{d_H(\mathbf{0},\mathbf{x}')}(1-p)^{n-d_H(\mathbf{0},\mathbf{x}')} \text{ changement de variable } \mathbf{x} + \mathbf{r} \rightarrow \mathbf{x}' \\
&= \sum_{\mathbf{x}' \in \mathcal{C} - \{\mathbf{0}\}} p^{d_H(\mathbf{0},\mathbf{x}')}(1-p)^{n-d_H(\mathbf{0},\mathbf{x}')} \\
&= \sum_{i=d}^{n} A_i p^i (1-p)^{n-i} \text{ En regroupant les mots de codes à la même distance de } \mathbf{0}
\end{aligned}
$$

La dernière égalité étant obtenue en remarquant que pour un code de distance minimale d, il n'existe pas de mot du code à une distance inférieure à d du mot de code nul (par définition de d_{min})