

# 1 Codage de canal

$$(n, k, d)_q$$

Lorsque la base est  $q = 2$  on ne l'affiche pas



|     |  |
|-----|--|
| $k$ | Nombre de bits à transmettre (information) |
| $n$ | Nombre de bits transmis $n \geq k$         |
| $d$ | Distance de Hamming                        |

Information (bits à transmettre)

$$x = (x_{k-1} \ x_{k-2} \ \cdots \ x_1 \ x_0) \in \mathbb{R}^{1 \times k}$$

Matrice systématique (identité à gauche).

$$G_S = (\overbrace{I_k}^{\text{blue}} \ \overbrace{P}^{\text{red}}) = \left( \begin{array}{cccc|ccc} 1 & 0 & \cdots & 0 & x & \cdots & x \\ 0 & 1 & \cdots & 0 & x & \cdots & x \\ \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & x & \cdots & x \end{array} \right) \in \mathbb{R}^{k \times n}$$

$$H_s = (\overbrace{P^T}^{\text{red}} \ \overbrace{I_{n-k}}^{\text{blue}}) \in \mathbb{R}^{n-k \times n}$$

Capacité de détection d'erreurs :  $d_{\min} - 1$   
 Capacité de correction d'erreurs :  $\left\lfloor \frac{d_{\min} - 1}{2} \right\rfloor$

## 1.1 Propriétés

$$G_S H_S^T = 0$$

## 1.2 Codes polynomiaux

1. Souvent linéaires (+ parfois cycliques)
2. Générateur rarement systématique (on peut utiliser un encodage systématique si le générateur ne l'est pas)

## 1.3 Codes linéaires

La somme de deux codes valides donne un nouveau code valide

## 1.4 Codes cycliques

Un décalage vers la gauche  $0011 \longrightarrow 0110$  ou vers la droite  $0011 \longrightarrow 1001$  donne un autre code valide.  
 Un décalage d'un mot-code de longueur  $n$  dans  $\mathbb{R}_n[X]$  est similaire à une multiplication par  $X$

### 1.4.1 Générateur

Un générateur permet, par des décalages cycliques et des sommes d'obtenir tous les mots-code.

Si le générateur divise  $(1 + X^n)$  alors le code est cyclique

### 1.4.2 Encodage

On veut encoder un message  $x$  et obtenir le mot-code  $y$

**Encodage systématique** :

$$x \xrightarrow{\text{encodage}} xX^{n-k} + \text{reste} \left( \frac{xX^{n-k}}{g(x)} \right) = y$$

Les bits du message se retrouvent au début du mot-code

**Encodage "standard"** :

$$x \xrightarrow{\text{encodage}} xg(x) = y$$

### 1.4.3 Vérification

On va simplement chercher à savoir si le message est correct ou non.

Vérification systématique (on vérifie simplement que le message soit égal au reste, comme pour l'encodage)

$$y[0 : k] == \text{reste} \left( \frac{y[0 : k]X^{n-k}}{g(x)} \right)$$

Vérification "standard"

$$s(x) = e(x) = \frac{y}{g(x)} \quad s(x) = 0 \longrightarrow \text{ok}$$

### 1.4.4 Correction d'erreurs

On va utiliser une matrice de contrôle (typiquement pour les codes BCH) qui va indiquer la position de(s) erreur(s)

### 1.4.5 Décodage

Dans le cas où le message est correct, on va effectuer le décodage.

Décodage systématique (on récupère simplement les bits dans le message tels-quels):

$$\hat{x} = y[0 : k]$$

Décodage "standard" :

$$\hat{x} = \frac{y}{g(x)}$$

## 1.5 Générateur $\leftrightarrow$ matrice

On peut créer la matrice en effectuant des décalages cycliques du générateur  $g(x)$

$$\begin{pmatrix} g_0 & g_1 & \cdots & g_{n-k} & 0 & 0 \\ 0 & g_0 & g_1 & \cdots & g_{n-k} & 0 \\ 0 & 0 & g_0 & g_1 & \cdots & g_{n-k} \end{pmatrix}_{k \times n}$$

Avec le nombre de lignes correspondant à la longueur du message. Cette matrice n'est pas systématique. Il faut la manipuler si on veut obtenir la version systématique. On effectue ensuite des combinaisons linéaires sur les lignes pour obtenir la matrice systématique.

## 1.6 Codage convolutionnel