Techniques et Technologies de Communication

Chapitre 3 : Chaine de transmission numérique

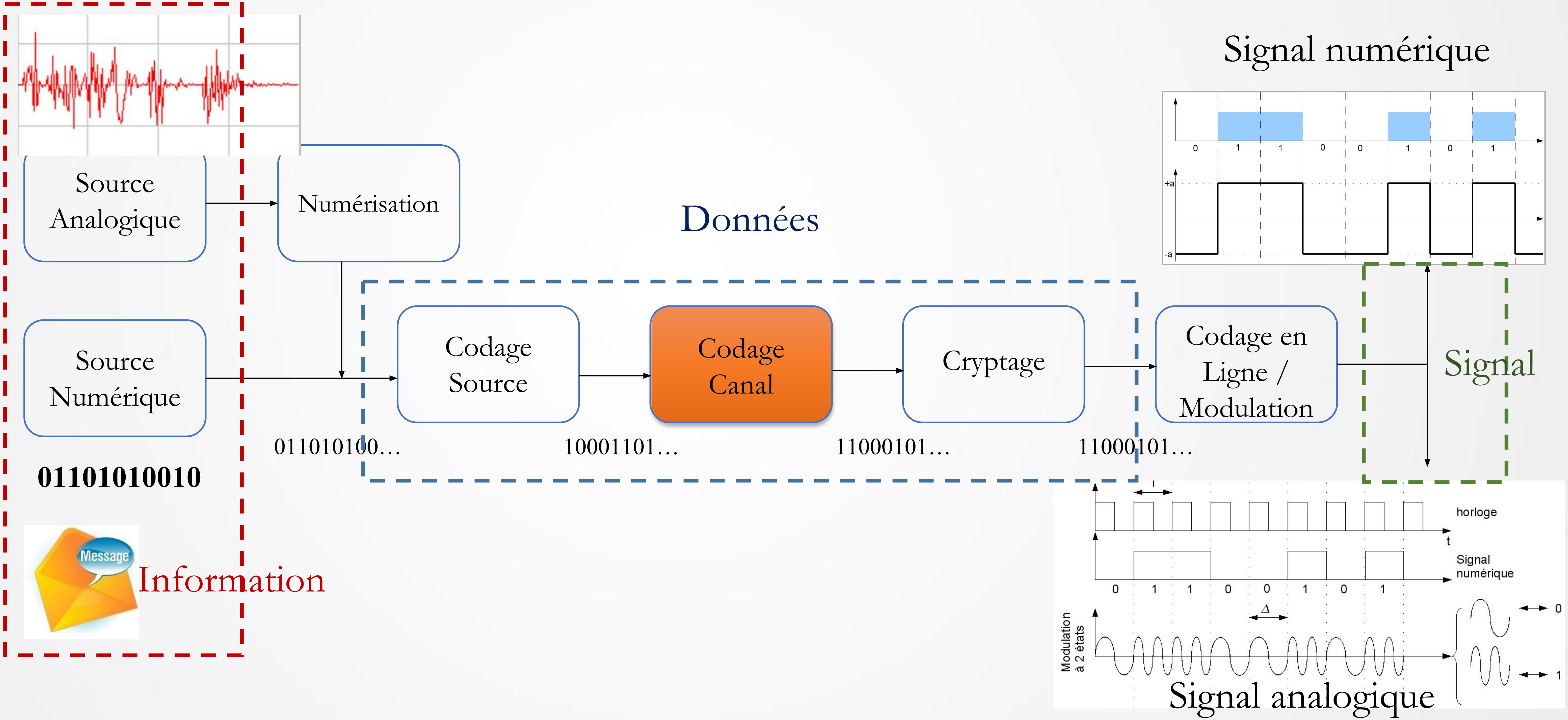
Rédigé par: Houda Khedher
Mis-à-jour par: équipe réseau-télécom

Sommaire

Chapitre 3 : Chaine de transmission numérique

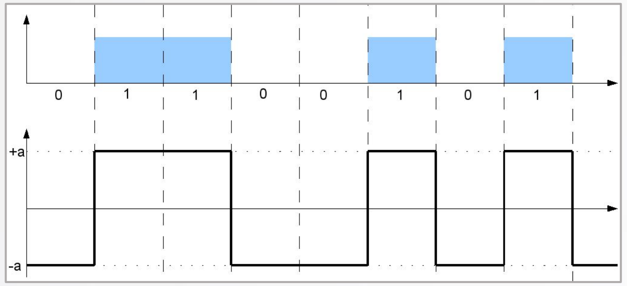
Partie 3 : Codage Canal

Chaine de transmission : Emission



Chaine de transmission : Réception

Signal numérique



Destination Analogique

Convertisseur numérique analogique

Données

Signal

Décodage en ligne / Démodulation

10001101…

Décryptage

11000101…

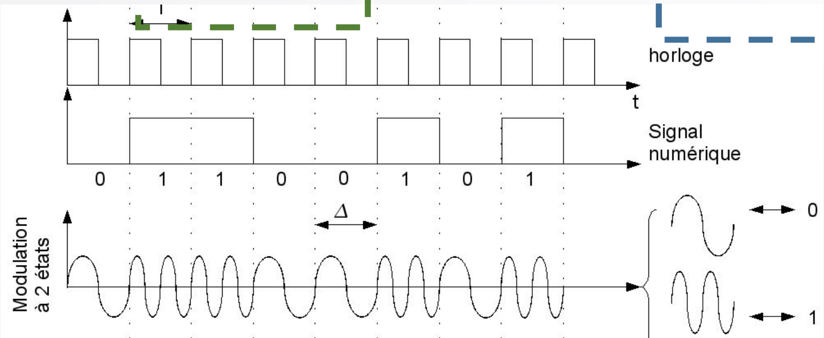
Décodage Canal

11000101…

Décodage Source

011010100…

Destination Numérique



Signal analogique

01101010010

Information

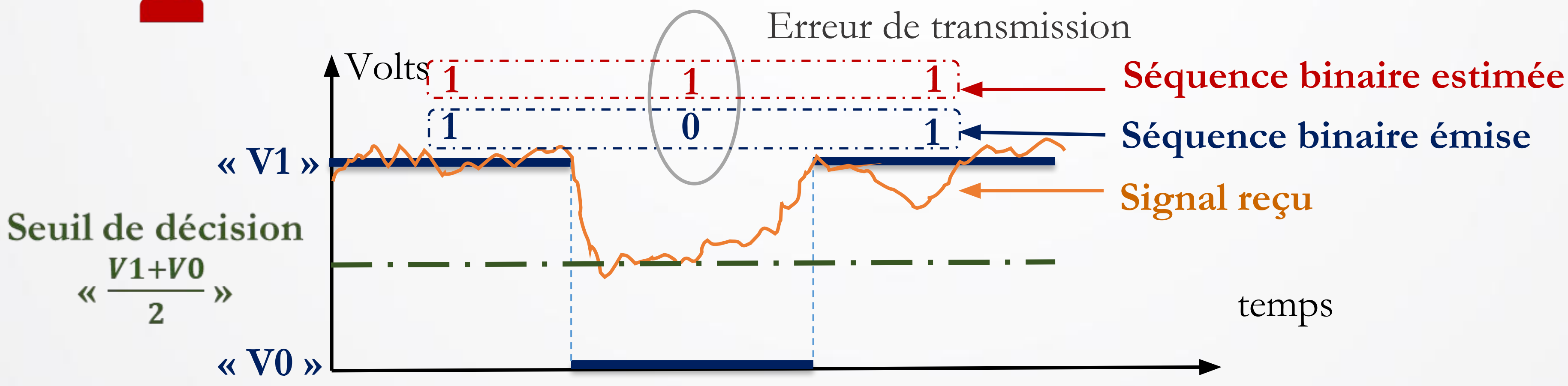
Codage Canal : Définition (1/4)

Objectif

Renforcer le signal contre les effets indésirables du canal de transmission

Différents phénomènes parasites (bruit, interférences, …) perturbent le canal de transmission et peuvent affecter les informations en modifiant un ou plusieurs bits du message transmis, introduisant ainsi des erreurs de transmission.

Qu'est ce qu'une erreur de transmission?



Règle d'estimation de la séquence binaire

Si le voltage reçu entre **V0** et $\frac{V1+V0}{2}$ le symbole reçu est **0** sinon c'est un **1**

Codage Canal : Définition (2/4)

Besoin : Contrôle des erreurs

Détection de l’erreur

On appelle détection d’erreur les mécanismes mis en œuvre pour que le système destinataire puisse vérifier l’intégrité (validité) des données reçues.

Attention, il y a des erreurs

Correction de l’erreur

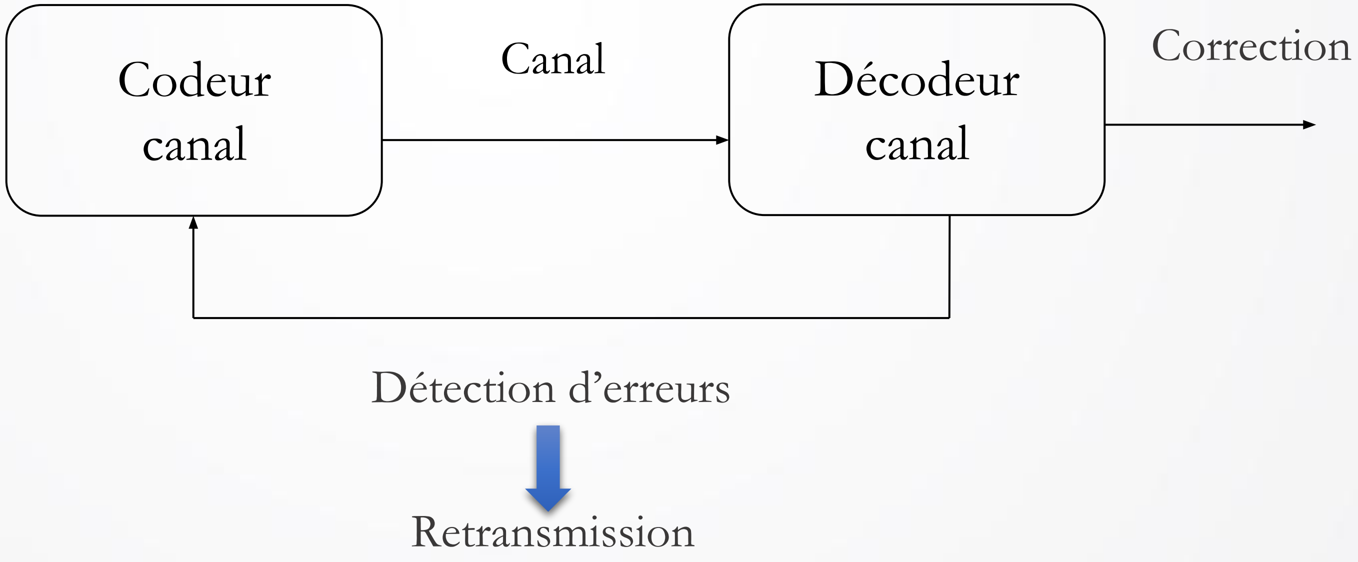
On appelle correction d’erreur les mécanismes mis en œuvre pour que le système destinataire puisse « corriger » les erreurs détectées dans la séquence reçue. La correction de l’erreur consiste à inverser le bit erroné.

L’erreur est au $i^{\text{ème}}$ bit

Codage Canal : Définition (3/4)

Besoin : Contrôle des erreurs

Différence entre détection et correction de l'erreur



Codage Canal : Définition (4/4)

Gagner en terme

Robustesse : signal moins sensible aux bruits et interférences

Qualité : moins d'erreurs

Energie : moins de retransmissions

Temps : moins de retransmissions

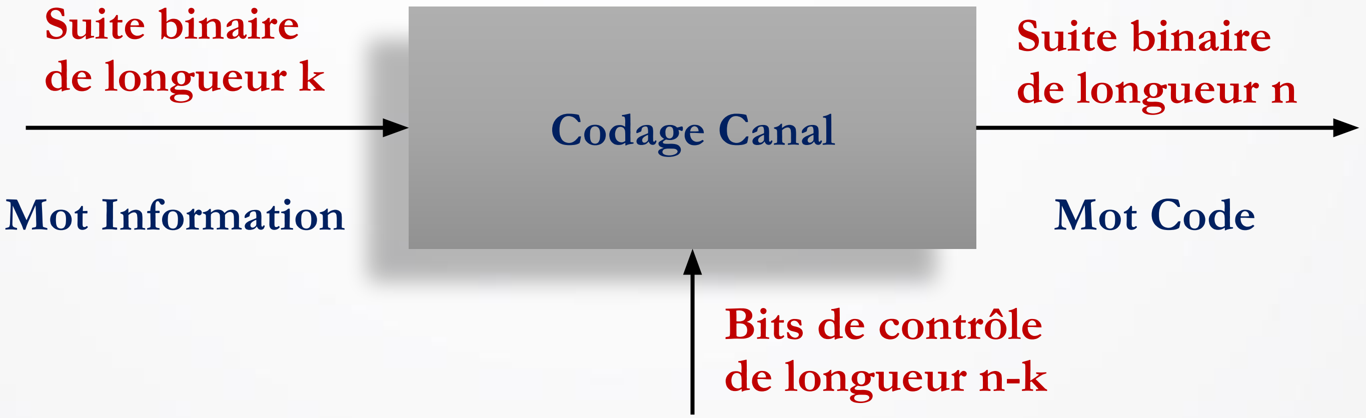
Codage Canal : Principe

Comment

Introduire de la <u>redondance utile</u> dans les données à l'entrée du codeur

→ des bits de contrôle d'erreurs

Introduire une redondance utile



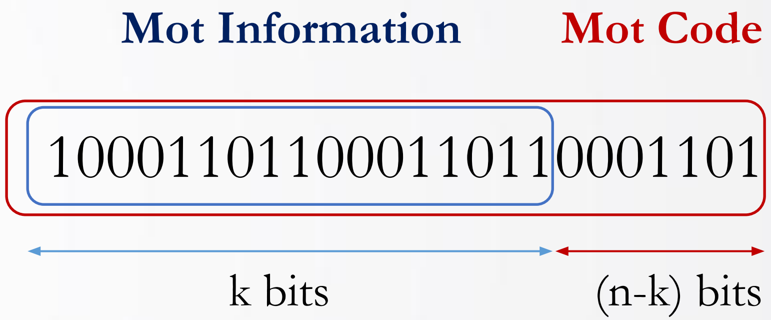
Avec, n>k

Codage Canal : Techniques

Deux familles de codes

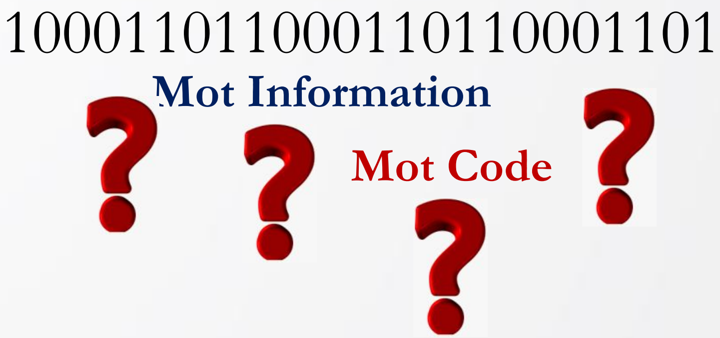
Codage Systématique

Le mot information peut être déduit directement du mot code (les bits du mot information se trouvent en des positions connues dans le mot code)



Codage non systématique

Le mot information et les bits de contrôle ne sont pas séparés



Code de parité (1/4)

Principe

- Ajouter, à la séquence binaire à protéger (mot information), <u>un bit</u>, telle que le nombre des bits à 1 transmis dans le mot code soit paire.

$n=k+1$

Exemple

Considérons la suite binaire 11000110. Le contrôle de parité s'applique pour chaque deux bits

➡ La suite binaire 11|00|01|10 est décomposée en 4 mots information

Mots Information		**Mots Code**
$k=2$		$n=3$
11	⟶	11**0**
00	⟶	00**0**
01	⟶	01**1**
10	⟶	10**1**

11000110 ⟶ **Code de parité** ⟶ 11**0**00**0**01**1**10**1**

Code de parité (2/4)

Différents schémas de codage

- Parité paire des bits à 1: le nombre total des ‘1’ est paire

  01 ⟶ 01**1**

- Parité paire des bits à 0: le nombre total des ‘0’ est paire

  01 ⟶ 01**0**

- Parité impaire des bits à 1: le nombre total des ‘1’ est impaire

  00 ⟶ 00**1**

- Parité impaire des bits à 0: le nombre total des ‘0’ est impaire

  00 ⟶ 00**0**

Code de parité (3/4)

Exemple

Nous souhaitons envoyer le message ‘HELLO’. La suite binaire correspondante est:

1001000 1000101 1001100 1001111 1000010

Appliquons le contrôle de parité à chaque mot de **7 bits** en utilisant la **parité paire des bits à 1** (k=7 bits et n=8 bits)

10010000 10001011 10011001 10011111 10000100

Suite à une transmission sur un canal bruité, la suite binaire reçue est:

10110000 10111011 10110011 10101001 10011011

1 erreur détectée mais pas localisée | **0 erreur détectée !** | **1 erreur détectée !** | **0 erreur détectée !** | **1 erreur détectée !**

Nbr de « 1 » impaire =>Détection d’erreurs

Nbr de « 1 » paire =>Pas de détection d’erreurs

Code de parité (4/4)

Caractéristiques du code

- ✔ Simple
- ✔ Code détecteur d'erreur
- ✘ L'erreur ne peut pas être localisée => **Pas de correction d'erreur**
- ✘ Peu efficace : pouvoir de détection limité

Nombre d'erreurs **paire** ⟶ **Aucune erreur n'est détectée !**

Nombre d'erreurs **impaire** et **>1** ⟶ **Une seule erreur est détectée !**

=> Plus le mot binaire est **long**, plus le pouvoir de détection est **limité**

Code CRC : Cyclic Redundancy Check

Principe

- Les bits de contrôle, appelés la clé CRC, sont déterminés par un ensemble d'opérations de l'arithmétique booléenne. La méthode de contrôle par clé CRC considère le bloc de K bits à transmettre comme un **polynôme de degré K −1 (P(x)).**
  Ce polynôme est divisé par un autre, dit **polynôme générateur G(x)**, Le reste de cette division constitue la clé CRC. **Le nombre de bits de contrôle** est égal au **degré du polynôme générateur**.

Etapes de codage

Etape 1 : Représenter le mot information un polynôme P(x).

Etape 2 : Multiplier le polynôme P(x), représentant le mot information, par le monôme du plus haut degré du polynôme générateur G(x).

Etape 3 : Diviser le polynôme résultant de l'étape 2 par polynôme générateur G(x).

Etape 4 : Coder le reste de division. Déduire le nombre de bits de contrôle.

Code CRC : Principe de codage (1/4)

Exemple

- On désire protéger le message « 110111 » par une clé calculée à l'aide du polynôme générateur $x^2 + x + 1$

Etape 1 : Représenter le mot information par un polynôme P(x).

Au message 110111, on fait correspondre le polynôme P(x):

On multiplie le monôme par la valeur du bit

1	1	0	1	1	1
Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0

$$P(x) = 1.x^5 + 1.x^4 + 0.x^3 + 1.x^2 + 1.x^1 + 1.x^0$$

$$P(x) = x^5 + x^4 + x^2 + x + 1$$

Code CRC : Principe de codage (2/4)

Exemple

- On désire protéger le message « 110111 » par une clé calculée à l'aide du polynôme générateur $x^2 + x + 1$

$$P(x) = x^5 + x^4 + x^2 + x + 1$$

Etape 2 : Multiplier le polynôme P(x), représentant le mot information, par le monôme du plus haut degré du polynôme générateur G(x).

$$G(x) = \boxed{x^2} + x + 1$$

Cette multiplication permet l'addition de la clé au message (décalage à gauche). Le dividende devient

$$P(x).x^2 = x^7 + x^6 + x^4 + x^3 + x^2$$

Code CRC : Principe de codage (3/4)

Exemple

- On désire protéger le message « 110111 » par une clé calculée à l'aide du polynôme générateur $x^2 + x + 1$

Etape 3 : Diviser le polynôme résultant de l'étape 2 par le polynôme générateur G(x).

$$
\begin{array}{l|l}
x^7+x^6+x^4+x^3+x^2 & x^2+x+1 \\
\underline{x^7+x^6+x^5} & \overline{x^5+x^3+1} \\
x^5+x^4+x^3+x^2 & \\
\underline{x^5+x^4+x^3} & \\
x^2 & \\
\underline{x^2+x+1} & \\
\boxed{x+1} &
\end{array}
$$

Reste de division

Remarque : En arithmétique booléenne, l'addition et la soustraction sont la même opération. Elles correspondent à l'opération logique OU Exclusif (XOR)

0+0=0
0+1=1
1+0=1
1+1=0

Code CRC : Principe de décodage

Etapes de codage

Etape 1 : Représenter le mot code reçu un polynôme R(x).

Etape 2 : Diviser le polynôme R(x) par le polynôme générateur G(x).

Etape 3 : Vérifier le reste de la division

Reste de division nul → ✔ **Pas d'erreur**

Reste de division non nul → ✘ Il y a au moins une erreur **<u>détectée</u>** qui ne peut pas être localisée
=> **<u>Pas de correction</u>**

Caractéristiques du code CRC

- Simple
- Code détecteur d'erreur : Pas de détermination du nombre d'erreur et pas de correction (sauf en cas d'utilisation d'un algorithme dédié)

Code CRC : Exemple de décodage (1/2)

Exemple 1

- On désire décoder la suite binaire reçue suivante « 11011111 » à l'aide du polynôme générateur x^2+x+1

Etape 1 : Représenter le mot code reçu un polynôme R(x).

$$R(x) = x^7 + x^6 + x^4 + x^3 + x^2 + x + 1$$

Etape 2 : Diviser le polynôme R(x) par le polynôme générateur G(x).

$$\begin{array}{l|l}
x^7+x^6+x^4+x^3+x^2+x+1 & x^2+x+1 \\ \cline{2-2}
\underline{x^7+x^6+x^5} & x^5+x^3+1 \\
x^5+x^4+x^3+x^2+x+1 & \\
\underline{x^5+x^4+x^3} & \\
x^2+x+1 & \\
\underline{x^2+x+1} & \\
\boxed{0} &
\end{array}$$

Reste de division

Reste de division nul ⟶ **Pas d'erreur**

Code CRC : Exemple de décodage (2/2)

Exemple 2

- On désire décoder la suite binaire reçue suivante « 11<u>1</u>11111 » à l'aide du polynôme générateur $x+1 = 1.x+1.1 \Rightarrow 11$

Etape 1 : Représenter le mot code reçu un polynôme R(x).

$$R(x) = x^7 + x^6 + x^5 + x^4 + x^3 + x^2 + x + 1$$

Etape 2 : Diviser le polynôme R(x) par le polynôme générateur G(x).

$$\begin{array}{l|l} x^7+x^6+x^5+x^4+x^3+x^2+x+1 & x^2+x+1 \\ \hline \underline{x^7+x^6+x^5} & x^5+x^2 \\ x^4+x^3+x^2+x+1 & \\ \underline{x^4+x^3+x^2} & \\ \boxed{x+1} & \end{array}$$

Reste de division

Reste de division non nul

$\downarrow$

Il y a au moins une erreur <u>**détectée**</u> qui ne peut pas être localisée

=> <u>**Pas de correction**</u>

Code CRC : Récapitulation

Emetteur: Codage

Représenter le mot information par un polynôme P(x).

Soit le polynôme générateur
G(x)= $x^n + x^{n-1} + \cdots + x + 1$.
Calculer la division suivante:

$$
\begin{array}{r|l}
x^n\,.\mathbf{P(x)} & \mathbf{G(x)} \\ \hline
\vdots & \cdots \\
\mathbf{r(x)} &
\end{array}
$$

Degrés de r(x)< Degrés de G(x)
n=Degrés de G(x)=longueur code CRC

<u>Autres exemples:</u>
Si n=3 et r(x)= x+1=0. $x^2 + 1.x + 1.1$
→ Code CRC= **011**
Si n=2 et r(x)=x=1.x+0.1 → Code CRC=**10**

Récepteur: Décodage

Représenter le mot de code reçu par un polynôme R(x).

Soit le polynôme générateur
G(x)= $x^n + x^{n-1} + \cdots + x + 1$.
Calculer la division suivante

$$
\begin{array}{r|l}
\mathbf{R(x)} & \mathbf{G(x)} \\ \hline
\vdots & \cdots \\
\mathbf{r(x)} &
\end{array}
$$

Si r(x)=0 → pas d'erreur
Si r(x)≠0 → il y a des erreurs

Remarque: Le code CRC correspond aux **<u>n</u>** derniers bits sur le mot de code reçu

Evaluation de performance du codage canal (1/2)

Rendement du codeur

Le rendement, ***R***, permet de mesurer le rapport entre la taille des données à l'entrée du codeur et la taille des données à la sortie.

Suite binaire de longueur k → **Codage Canal** → Suite binaire de longueur n

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

Plus le rendement est élevé, moins de bits de contrôle sont insérés => Débit utile est élevé! On cherche ainsi à utiliser des codeurs à rendement ***R*** élevé.

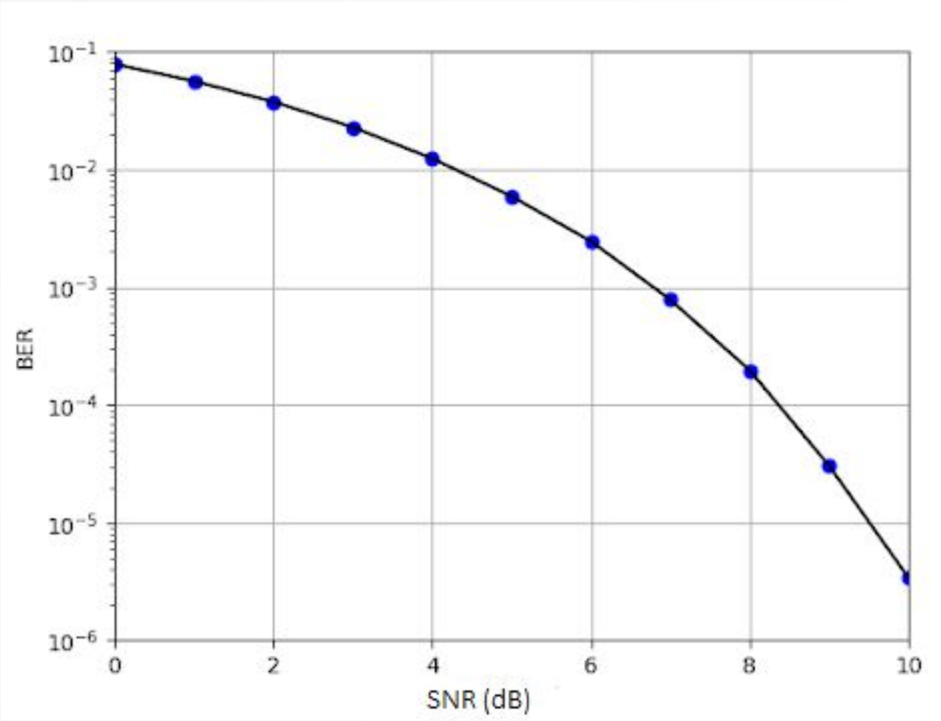
Hors, si les <u>perturbations dues au canal sont importantes</u>, il faudra ajouter plus de bits de contrôle et donc utiliser des codeurs à rendement ***R*** faible. Cela permettra la minimisation des erreurs et l'amélioration de la qualité.

Plus de qualité (moins d'erreurs)

Evaluation de performance du codage canal (2/2)

Taux d'erreur binaire

Le Taux d'erreur binaire (**TEB**), ou Bit Error Rate (**BER**) représente le rapport entre le nombre de bits erronés et le nombre total de bit transmis,



$$BER = \frac{nombre\ de\ bits\ erronés}{nombre\ total\ de\ bits\ transmis}$$

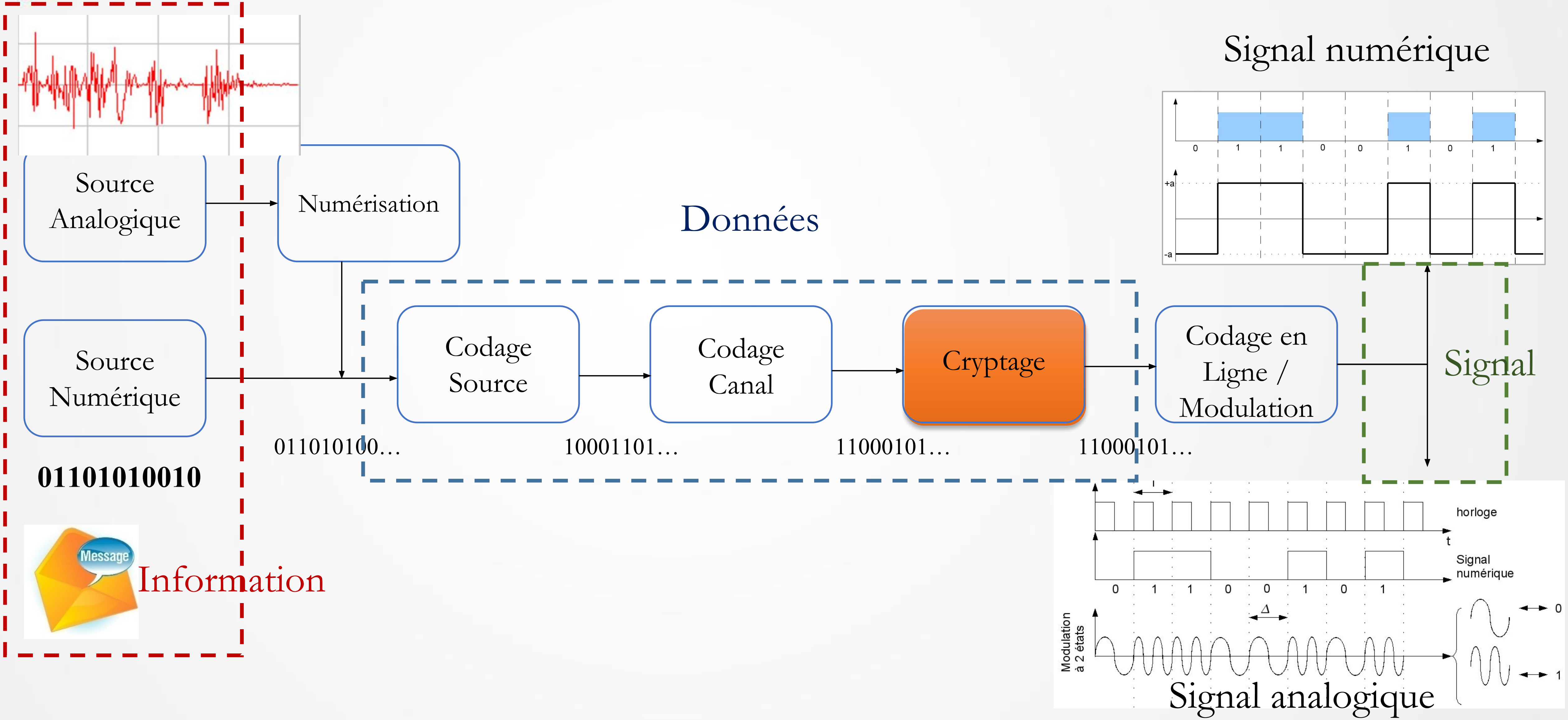
Un TEB faible signifie une meilleure qualité grâce à un meilleur contrôle d'erreur.

Plus de qualité (moins d'erreurs)

Chapitre 3 : Chaine de transmission numérique

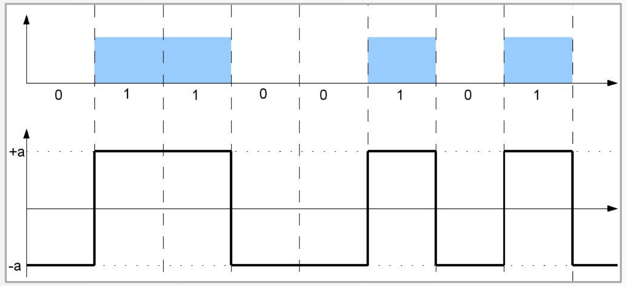
Cryptage

Chaine de transmission : Emission



Chaine de transmission : Réception

Signal numérique



Données

Signal

Décodage en ligne / Démodulation → Décryptage → Décodage Canal → Décodage Source

10001101…

11000101…

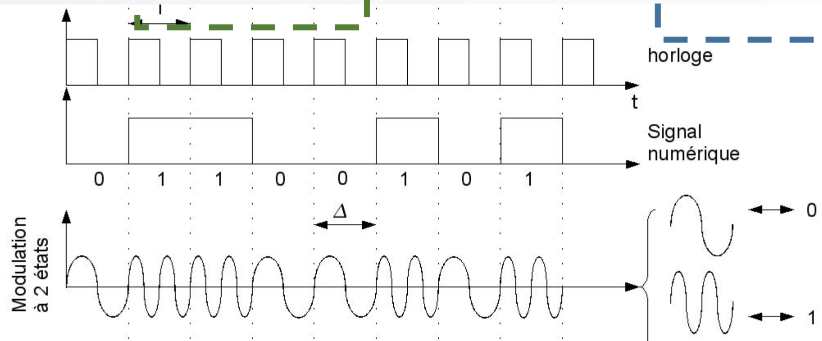
11000101…

Destination Analogique

Convertisseur numérique analogique

011010100…

Destination Numérique



Signal analogique

01101010010

Information

Cryptage

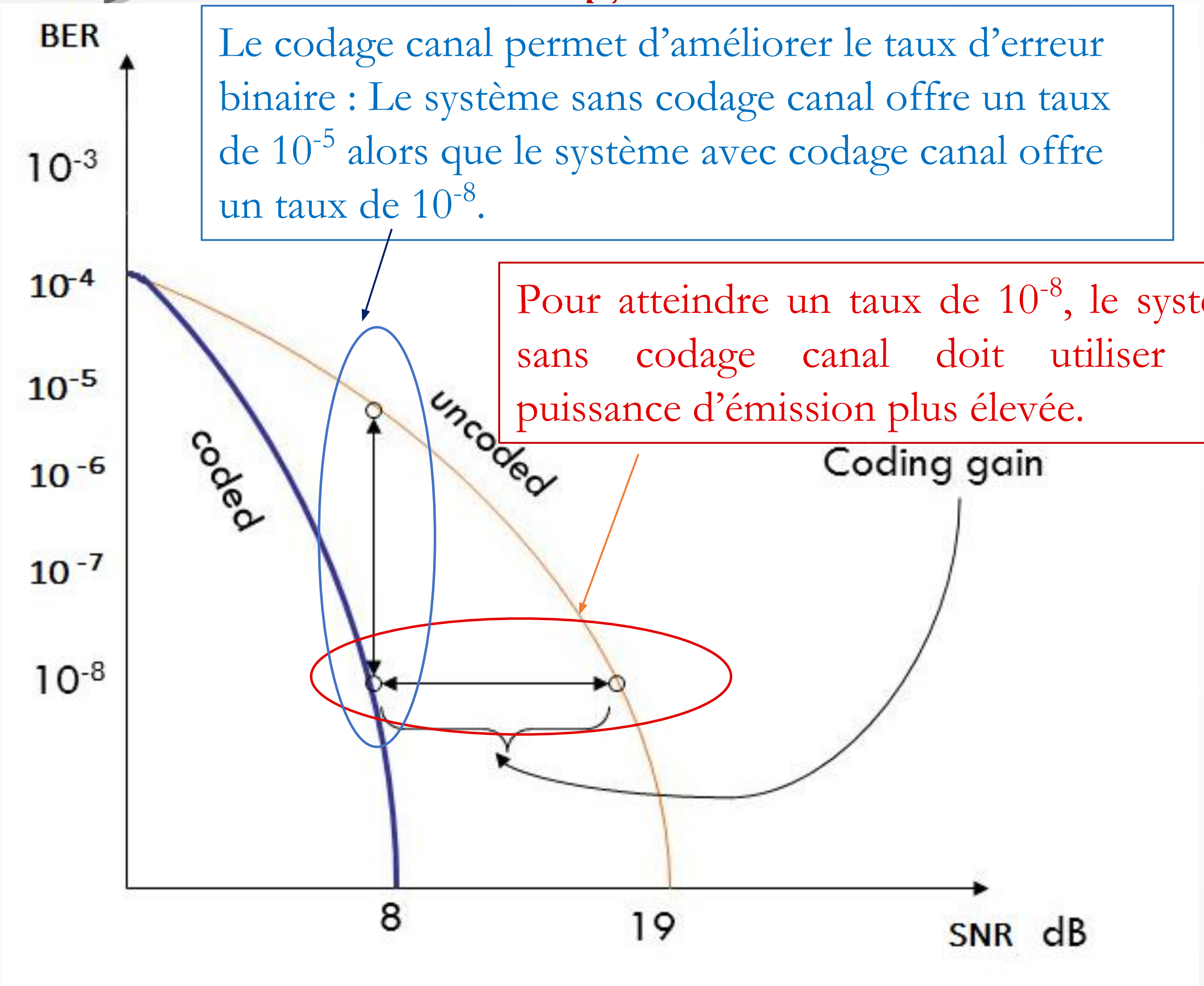
Objectif

Protéger les informations par des techniques de cryptographie afin d’empêcher des tierces personnes d'accéder ou de modifier les données.

Merci pour votre attention

Evaluation de performance du codage canal (2/2)

Gain de codage & Taux d'erreur binaire



Le codage canal permet d'améliorer le taux d'erreur binaire : Le système sans codage canal offre un taux de 10^{-5} alors que le système avec codage canal offre un taux de 10^{-8}.

Pour atteindre un taux de 10^{-8}, le système sans codage canal doit utiliser une puissance d'émission plus élevée.

Le Taux d'erreur binaire (**TEB**), ou Bit Error Rate (**BER**) représente le rapport entre le nombre de bits erronés et le nombre total de bit transmis,

$$BER = \frac{nombre\ de\ bits\ erronés}{nombre\ total\ de\ bits\ transmis}$$

Plus de qualité (moins d'erreurs)

Moins de consommation d'énergie