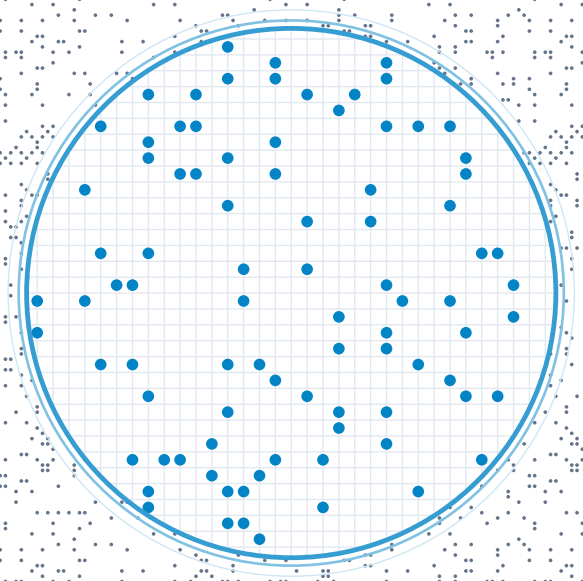
Codes LDPC

Anthony PERRONI

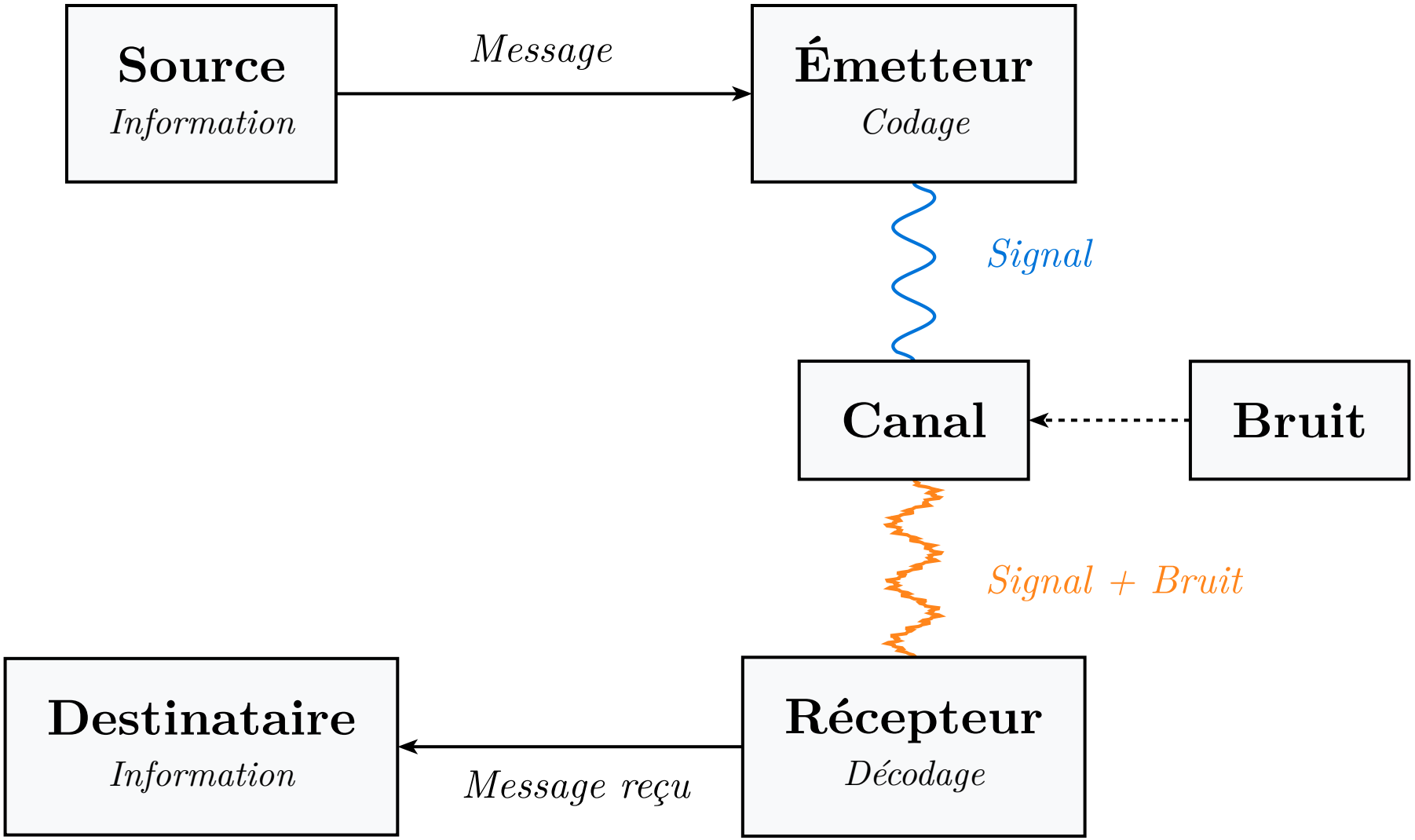
n°49871

2025 - 2026

Plan

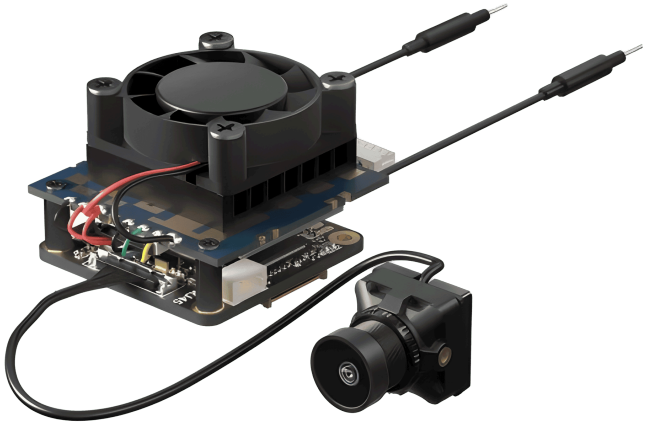


Introduction : Communication Numérique



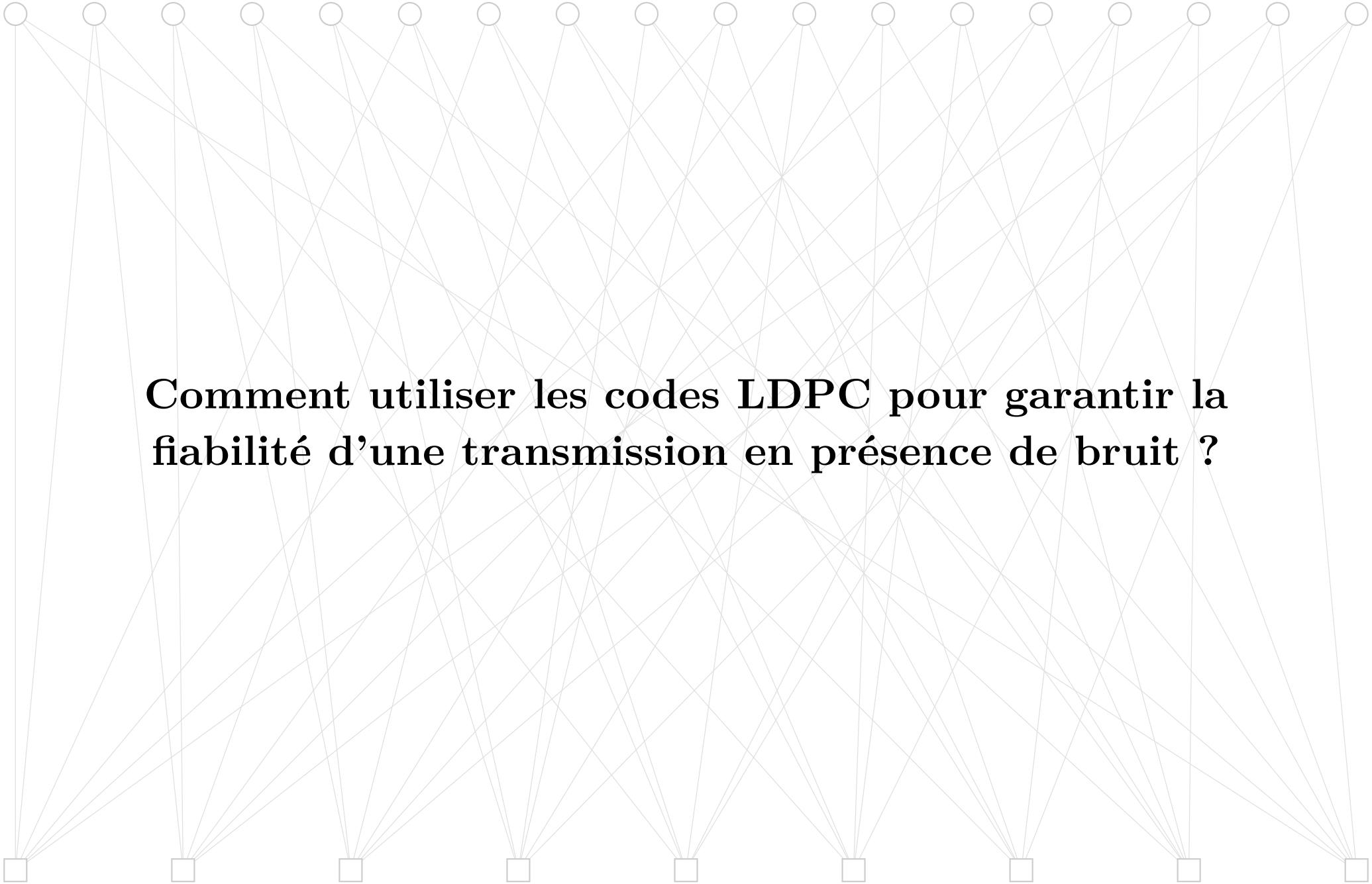
Introduction : Utilisation

Athena-Fidus



Module OpenIPC

Problématique



Comment utiliser les codes LDPC pour garantir la fiabilité d'une transmission en présence de bruit ?

Définition : Codes Linéaires en Bloc

Code $(n, k) \in \mathbb{N}^2$

$\mathcal{C}$ sous-espace vectoriel de dimension k de $\mathbb{F}_2^n$

- k : longueur du message original
- n : longueur du mot de code
- $m = n - k$: nombre de bits de parités

Encodage

$\Phi : \mathbb{F}_2^k \to \mathbb{F}_2^n \in \mathcal{L}\left(\mathbb{F}_2^k, \mathbb{F}_2^n\right)$

Définition : Codes Linéaires en Bloc

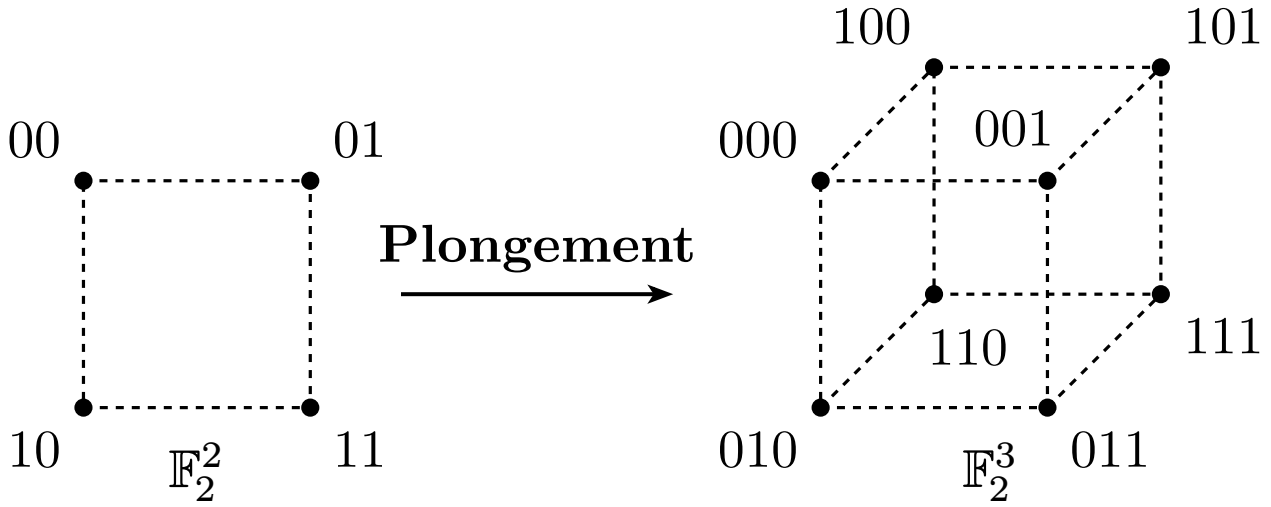
Code $(n, k) \in \mathbb{N}^2$

$\mathcal{C}$ sous-espace vectoriel de dimension k de $\mathbb{F}_2^n$

- k : longueur du message original
- n : longueur du mot de code
- $m = n - k$: nombre de bits de parités

Encodage

$\Phi : \mathbb{F}_2^k \to \mathbb{F}_2^n \in \mathcal{L}\left(\mathbb{F}_2^k, \mathbb{F}_2^n\right)$



Définition : Matrice Génératrice

Matrice Génératrice

$G \in \mathcal{M}_{k,n}(\mathbb{F}_2)$ dont les lignes sont une base de $\mathcal{C}$

Encodage

Pour un message $u \in \mathbb{F}_2^k$ le mot de code $c \in \mathcal{C}$ est :

$$c = \Phi(u) = u \odot G$$

Forme systématique

$$G = \left[\begin{array}{c|c} I_k & P \end{array}\right]$$

- Pour $u \in \mathbb{F}_2^k$, $\quad u \odot G = \left[\begin{array}{c|c} u & u \odot P \end{array}\right]$
- $P \in \mathcal{M}_{k,n-k}(\mathbb{F}_2)$ matrice de parité

Définition : Matrice de Contrôle

Matrice de Contrôle

$$H = \left[\begin{array}{c|c} P^\top & I_{n-k} \end{array}\right]$$

- $\mathcal{C} = \ker(H) = \{v \in \mathbb{F}_2^n \mid H \odot v^\top = 0\}$
- $G \odot H^\top = 0$

Syndrome

Pour un vecteur reçu $r = c + e,\ s \in \mathbb{F}_2^{n-k}$

$$s = Hr^\top = Hc^\top + He^\top = 0 + He^\top$$

- Si $s = 0,\ r$ est un mot de code valide
- Sinon s donne la signature de l'erreur e

Exemple d'un code linéaire

Exemple d'un code $(5, 2)$

- On choisit la matrice de parité P :

$$P = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}$$

- Alors la matrice génératrice G est :

$$G = \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{array}\right]$$

- Message $u = [1 \;\; 1]$
- Mot de code $c = uG$:

$$c = [1 \;\; 1] \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 \end{array}\right] = [1 \;\; 1 \;\; 1 \;\; 0 \;\; 1]$$

Exemple d'une code linéaire

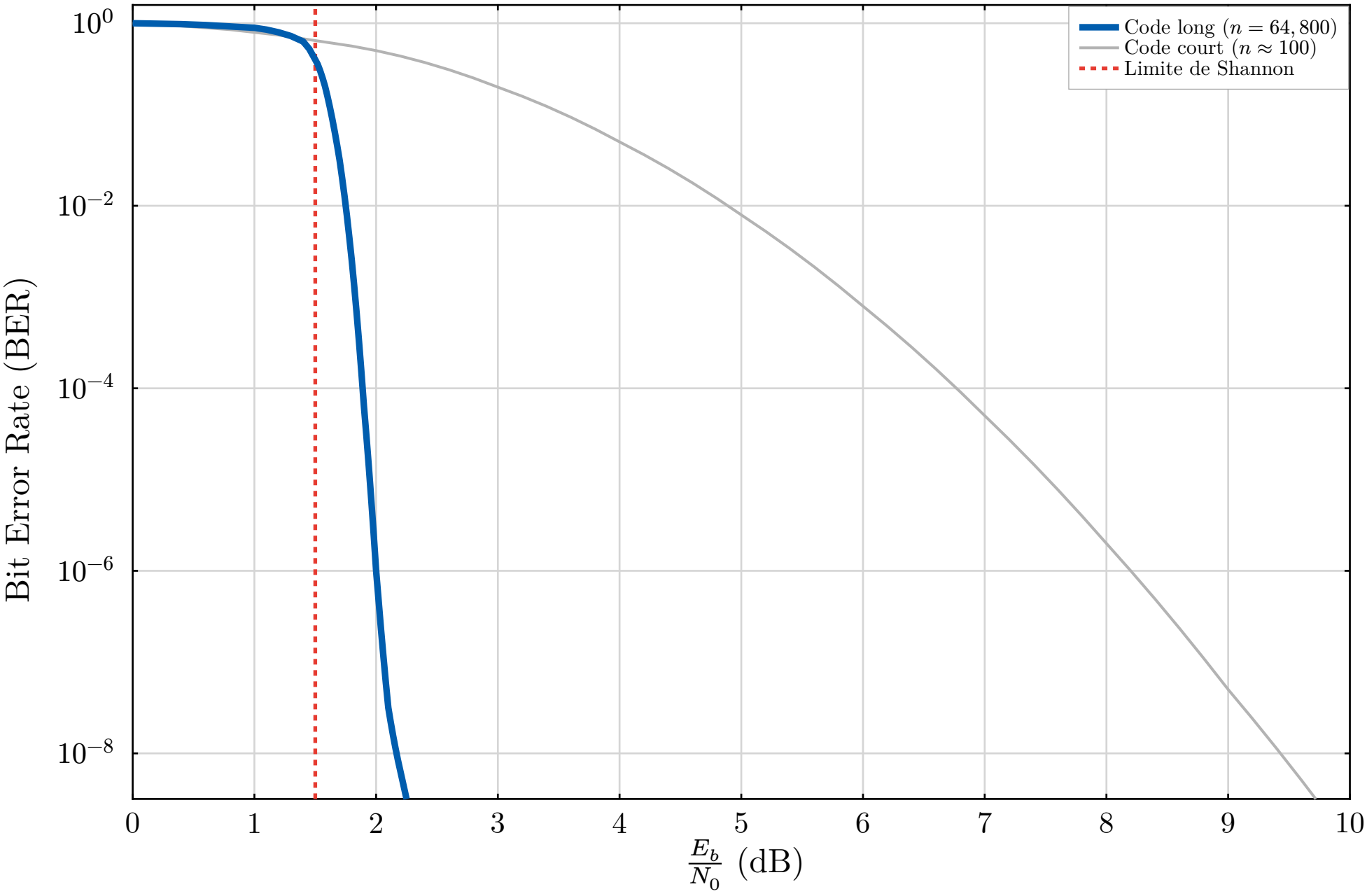
Enfin

$$H = \left[\begin{array}{cc|ccc} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{array}\right]$$

Vérification du mot de code $c = [1\ 1\ 1\ 0\ 1]$

$$Hc^\top = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{bmatrix} [1\ 1\ 1\ 0\ 1]^\top = \begin{bmatrix} 1 \oplus 0 \oplus 1 \oplus 0 \oplus 0 \\ 1 \oplus 1 \oplus 0 \oplus 0 \oplus 0 \\ 0 \oplus 1 \oplus 0 \oplus 0 \oplus 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

Approcher la Limite de Shannon



Le Mur de la Complexité

Décodage par Maximum de Vraisemblance (MDL)

Chercher le mot de code $\boldsymbol{c} \in \mathcal{C}$ le plus probable sachant $\boldsymbol{r}$ reçu :

$$\hat{\boldsymbol{c}} = \arg\min_{\boldsymbol{c} \in \mathcal{C}} d_H(\boldsymbol{r}, \boldsymbol{c})$$

- Équivalent à chercher l'erreur $\boldsymbol{e}$ de poids minimal tel que $\boldsymbol{H}\boldsymbol{e}^\top = \boldsymbol{s}$.

Le Problème du décodage par Syndrome

NP-Difficile et pour H quelconque : $\mathcal{O}\left(2^k\right)$

- Pour $k = 100$ bits, $2^{100} \approx 10^{30}$ opérations nécessaires.

Définition des Codes LDPC

Formalisation des Codes LDPC Réguliers

Code linéaire en bloc avec une matrice de contrôle $\boldsymbol{H}$ est **clairsemée**.

- **Poids de Colonne w_c**
- **Poids de Ligne w_r**

Conditions de Faible Densité

$$w_c \ll n - k \qquad w_r \ll n$$

Rendement

$$R = \frac{n - \mathrm{rg}(H)}{n} \geq 1 - \frac{m}{n}$$

Matrice de contrôle

Code LDPC (6, 3)

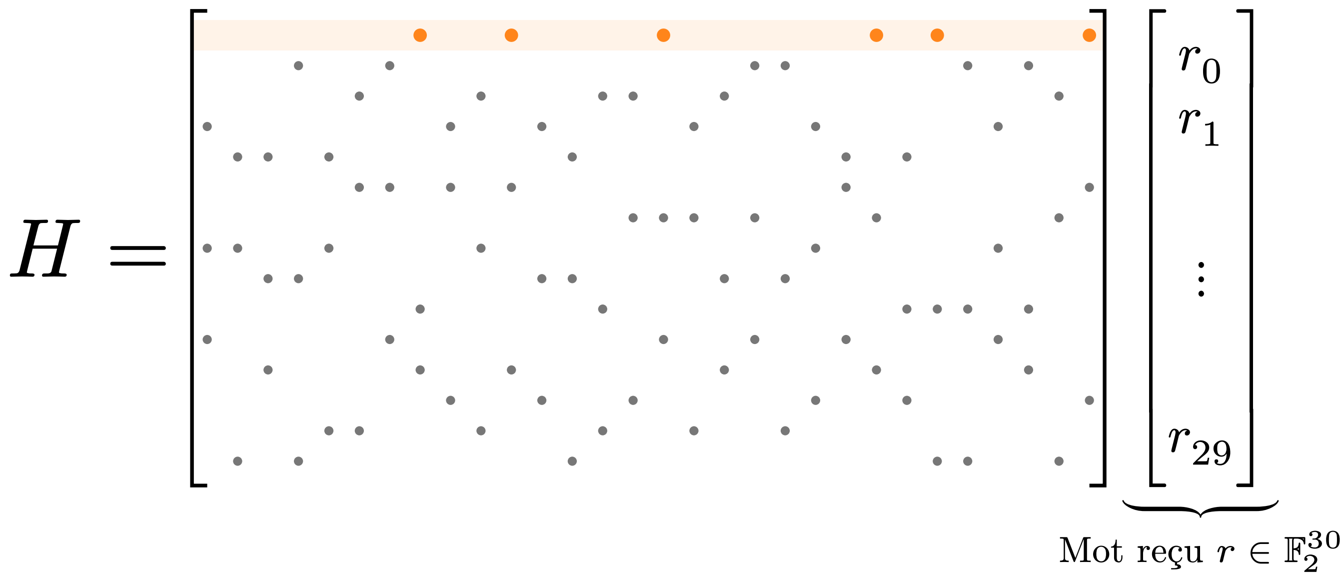
$mw_r = nw_c$ donc $H \in \mathcal{M}_{15,30}(\mathbb{F}_2)$ et $R = 1 - \frac{m}{n} = \frac{1}{2}$

$$H = \begin{bmatrix} & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \\ & & & & & & & & & & & & & & \end{bmatrix}$$

$w_c = 3$

$w_r = 6$

De la Matrice aux Équations de Parité



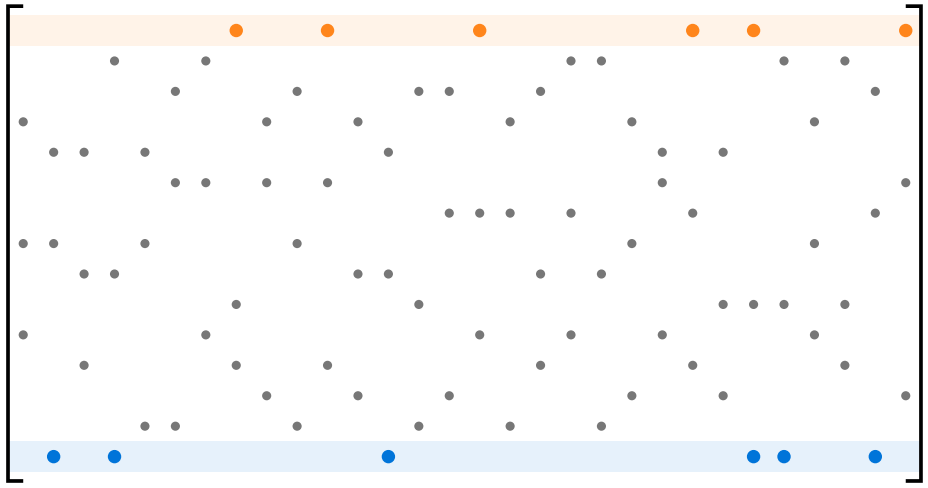
- Chaque ligne j de H définit une équation de parité f_j.
- Pour r, on vérifie le syndrome : $Hr^\top = 0$.

Équations de Parité

$$f_0 : r_7 \oplus r_{10} \oplus r_{15} \oplus r_{22} \oplus r_{24} \oplus r_{29} = 0$$

- Si $f_j = 1$, un nombre impair de bits a été inversé par le canal.

L'Entrelacement des Contraintes

$$H = \left[\right.$$  $$\left. \right]$$

- Chaque bit r_i participe à $w_c = 3$ équations distinctes

$$\begin{cases} r_7 \oplus r_{10} \oplus r_{15} \oplus r_{22} \oplus \boldsymbol{r_{24}} \oplus r_{29} = 0 \\ \vdots \\ r_1 \oplus r_3 \oplus r_{12} \oplus \boldsymbol{r_{24}} \oplus r_{25} \oplus r_{28} = 0 \end{cases}$$

- $\boldsymbol{r_{24}}$: Surveillé par f_0 et f_{14}.
- Si $f_0 = 1$ et $f_{14} = 1$, r_{24} est suspect

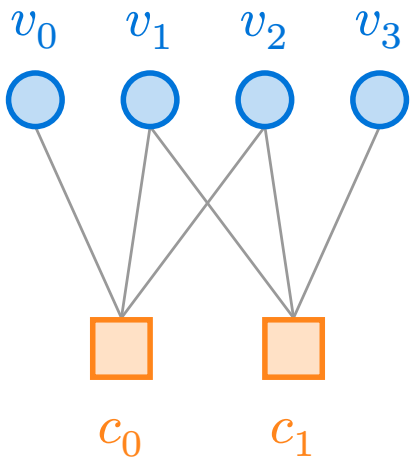
Graphe de Tanner : Définition

Graphe de Tanner $\mathcal{G}(H)$

Graphe bipartite $\mathcal{G} = (V \sqcup C, E)$:

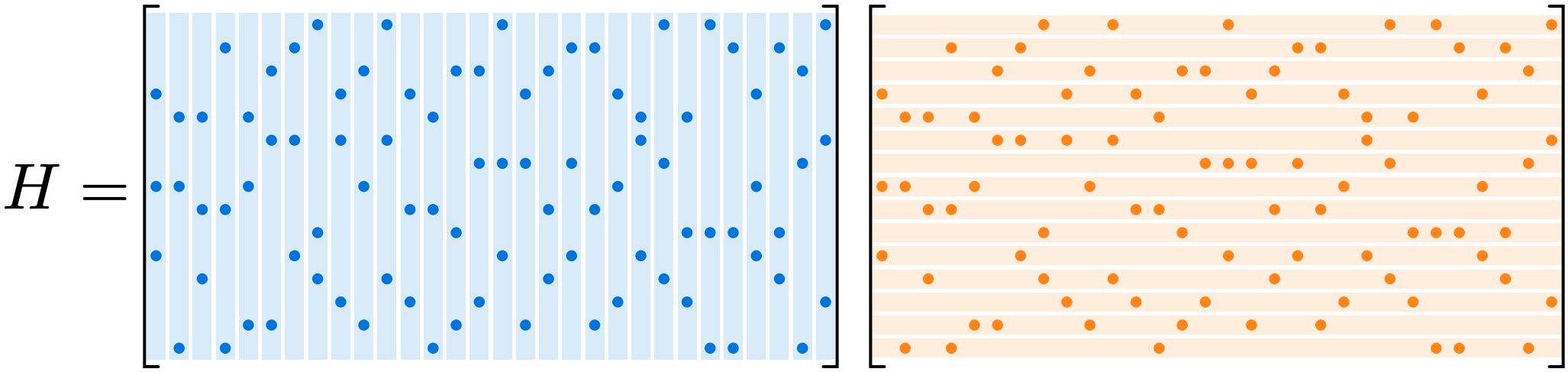
$$(v_j, c_i) \in E \iff H_{i,j} = 1$$

- $V = \{v_0, ..., v_{n-1}\}$ nœuds de **variable**
- $C = \{c_0, ..., c_{m-1}\}$ nœuds de **contrôle**
- $\deg(v_j) = w_c$
- $|E| = n \cdot w_c = m \cdot w_r$
- $H \cong \mathcal{G}$
- $\deg(c_i) = w_r$

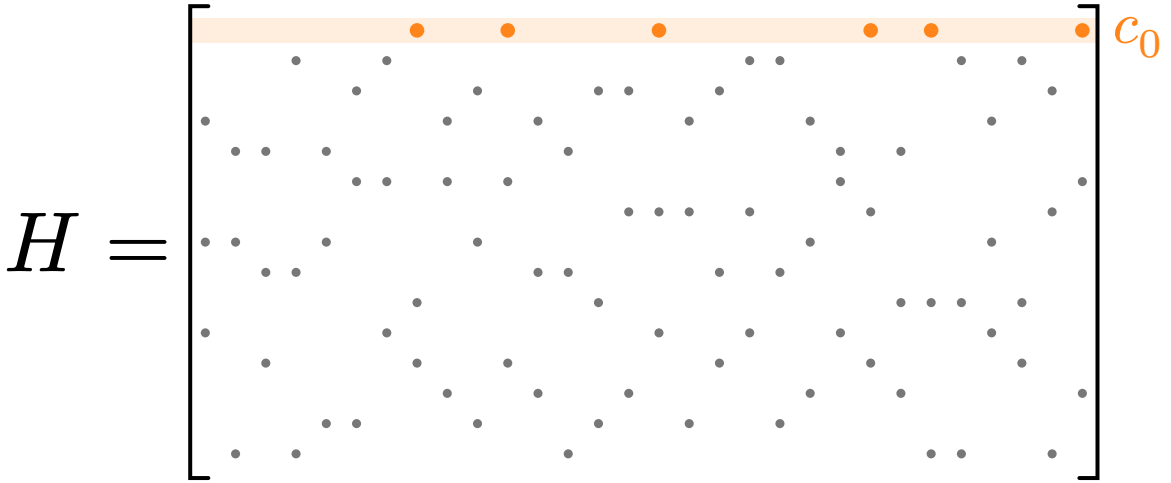


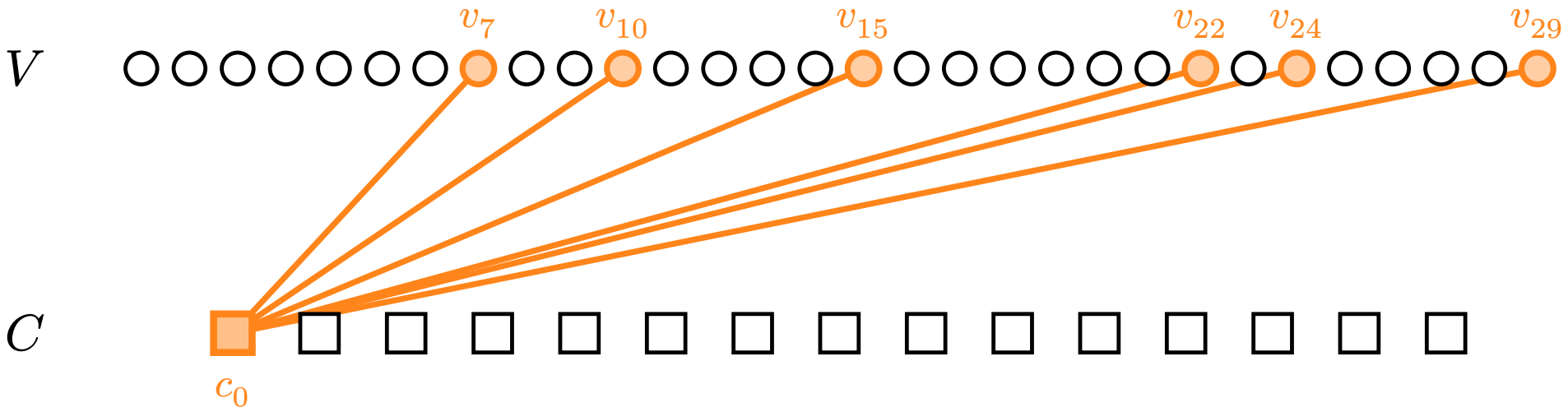
Exemple $n = 4,\ m = 2$

Construction du Graphe : Les Nœuds



Construction du Graphe : Nœud de Contrôle

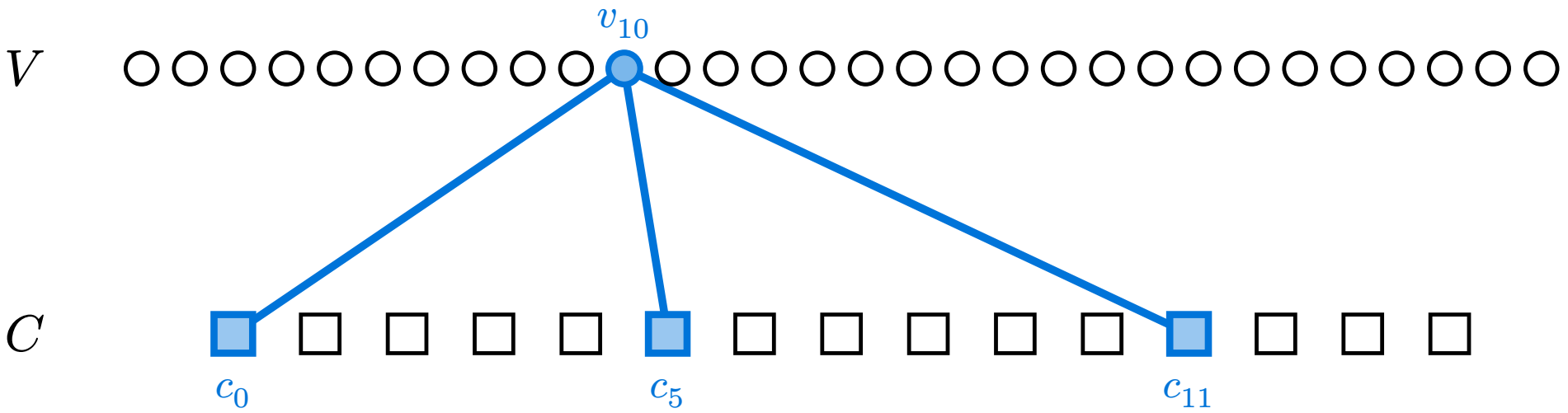




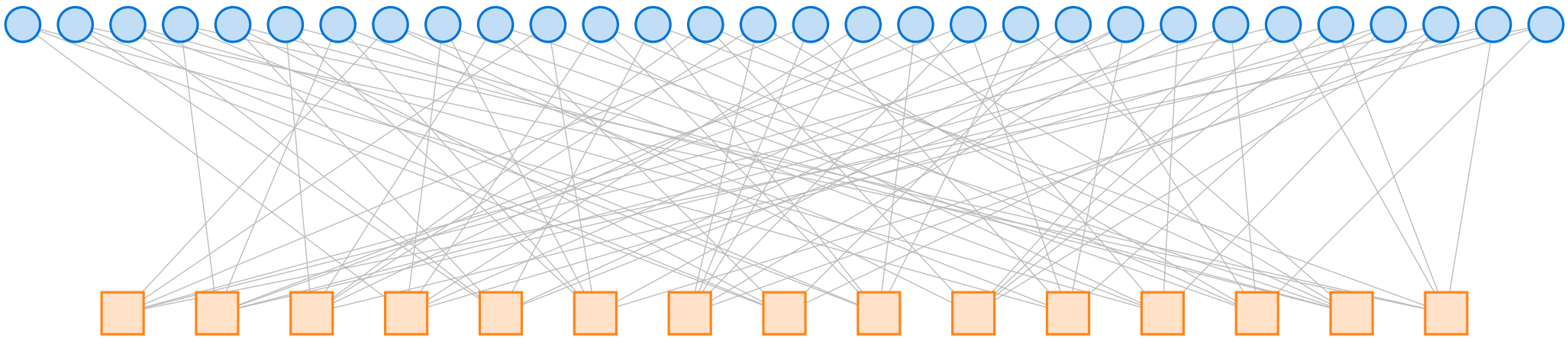
Construction du Graphe : Nœud de Variable

$$H = \left[\begin{array}{c} \\ \\ \\ \\ \\ \end{array}\right]$$

v_{10}



Graphe de Tanner Final



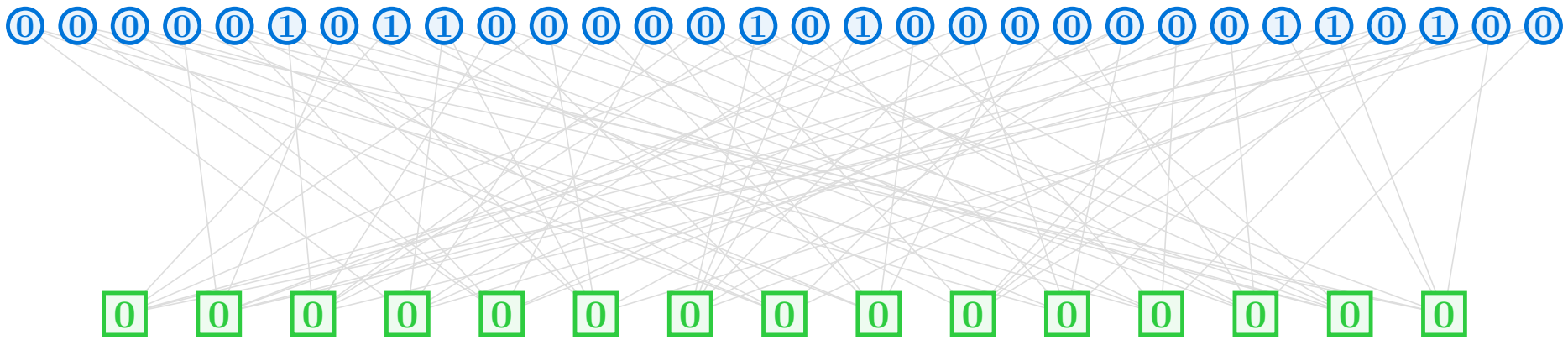
○ Nœuds de variable

□ Nœuds de contrôle

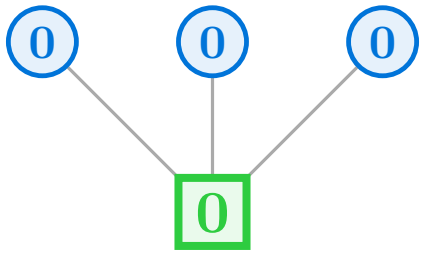
La Contrainte de Somme Nulle

Vision Graphe

Si $s = 0$ alors que chaque nœud de contrôle est localement satisfait



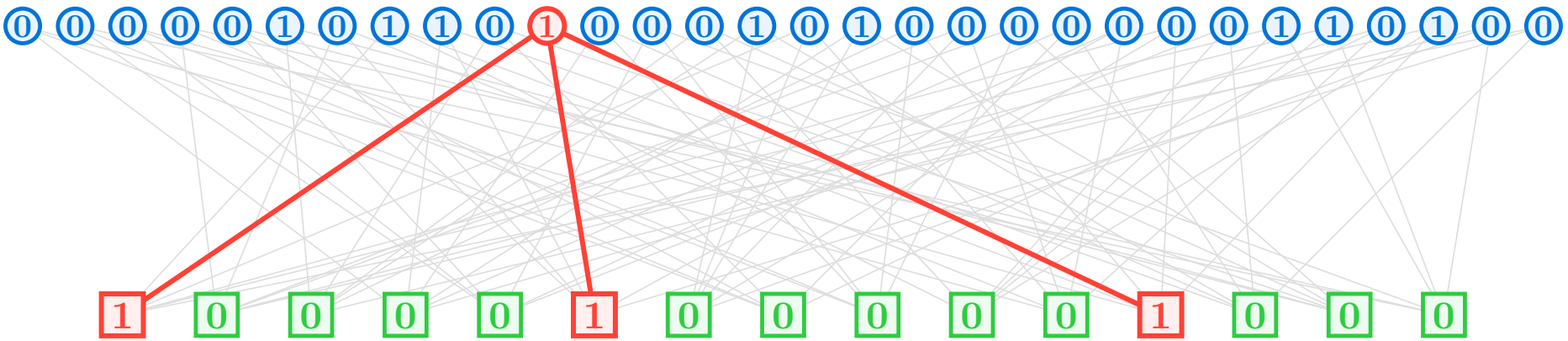
Chaque ▢ calcule le xor de ses voisins ○ : $f_i = \bigoplus_{j \in \mathcal{N}(c_i)} v_j$



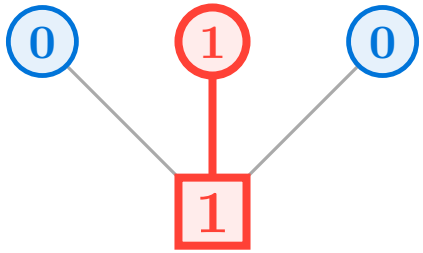
$0 \oplus 0 \oplus 0 = \mathbf{0}$

Détection d'Erreur

Si un bit est inversé, toutes les contraintes associées sont à 1



$$0 \oplus \mathbf{1} \oplus 0 = \mathbf{1} \rightarrow \textbf{Erreur détectée}$$



$$0 \oplus 1 \oplus 0 = 1$$

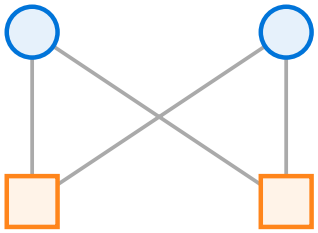
La Topologie de H : Le Girth

Définition : Le Girth (La Maille)

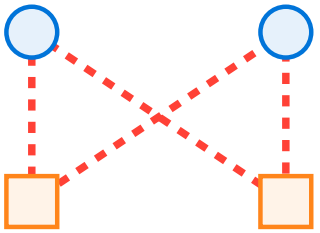
Longueur du plus court cycle dans le graphe de Tanner

- Le girth est **pair**
- La valeur minimale est $g = 4$.

Girth élevé ⇒ Meilleure diffusion de l'information.



Graphe de Tanner

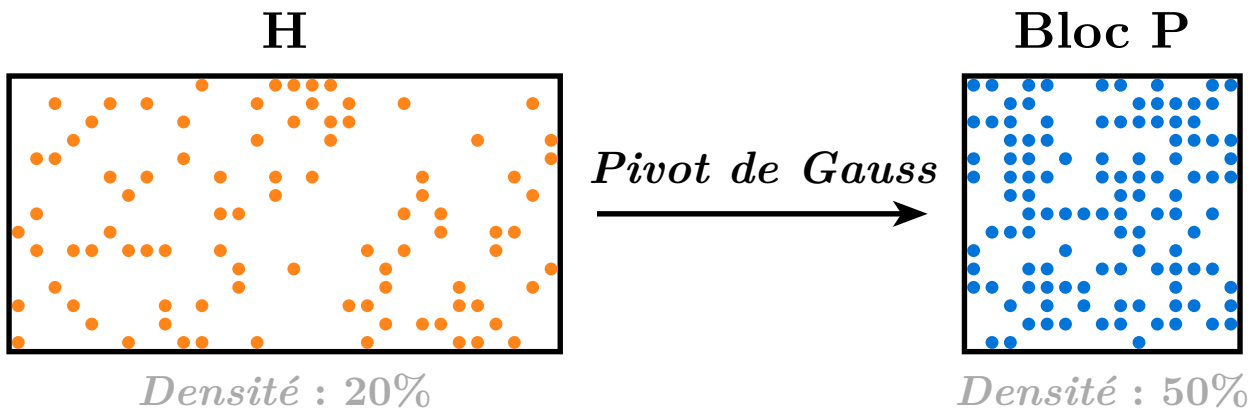


4-Cycle

Encodage LDPC : Calcul de G

Encodage

Mot de code $\boldsymbol{c}$ généré à partir d'un message $\boldsymbol{u}$: $\boldsymbol{c} = \boldsymbol{uG}$



- Forme Systématique : Par élimination de Gauss sur $\boldsymbol{H}$, on obtient

$$\boldsymbol{H} = \left[\, \boldsymbol{P}^\top \mid \boldsymbol{I}_{n-k} \,\right] \longrightarrow \boldsymbol{G} = \left[\, \boldsymbol{I}_k \mid \boldsymbol{P} \,\right]$$

- La matrice $\boldsymbol{G}$ devient dense $\Rightarrow$ encodage en $\mathcal{O}(n^2)$

Décodage : Bit-Flipping

Décision Stricte (Hard Decision)

Algorithme **itératif** : les nœuds **échangent des bits** pour localiser les erreurs.

Message Passing

- ○ envoie son bit courant à ses voisins □
- □ renvoie son **verdict de parité** (0 ou 1)

- Si v_j participe à **trop d'équations non satisfaites** $\Rightarrow$ on l'inverse.

Décodage : Bit-Flipping

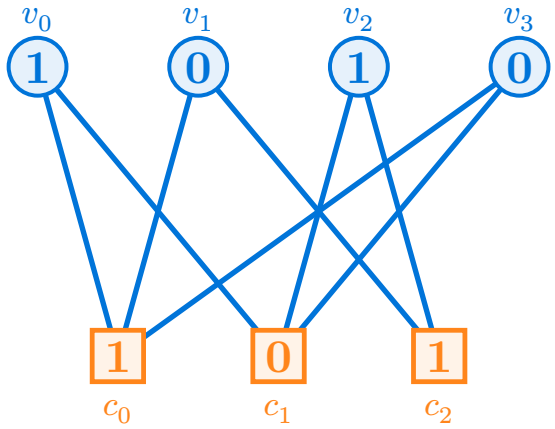
Décision Stricte (Hard Decision)

Algorithme **itératif** : les nœuds **échangent des bits** pour localiser les erreurs.

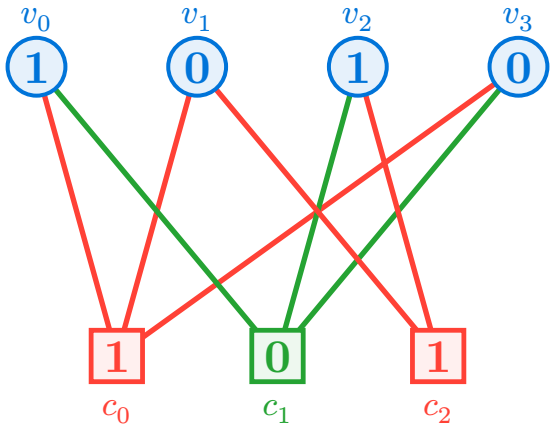
Message Passing

- ○ envoie son bit courant à ses voisins □
- □ renvoie son **verdict de parité** (0 ou 1)

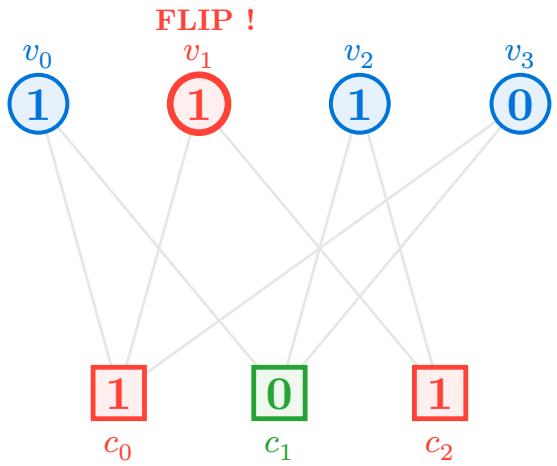
- Si v_j participe à **trop d'équations non satisfaites** $\Rightarrow$ on l'inverse.



VN → CN

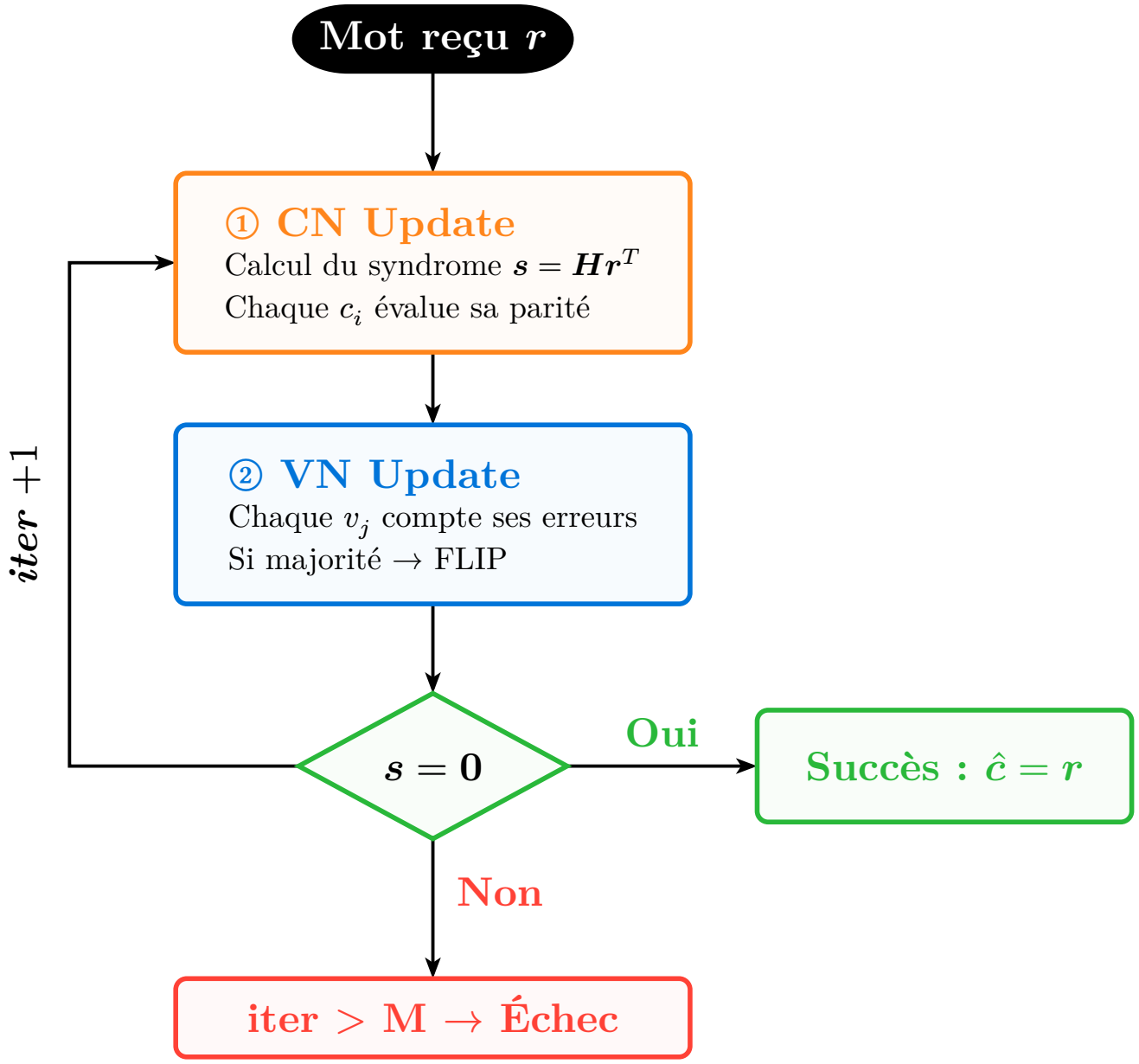


CN → VN



Correction

Bit-Flipping : Graphe de flot de contrôle



Exemple implementation Bit-Flipping rust

ex + canal d'etude bruit AWGN avec ce qu'il se passe dans les radio / cable etc + tension

Bit-Flipping : Analyse

Avantages

- **Complexité** : simples XOR et compteurs — $\mathcal{O}(n)$ par itération

Limite

- Ignore la **confiance** du récepteur physique dans le signal
- Un bit reçu à 0.51 V est traité comme 0

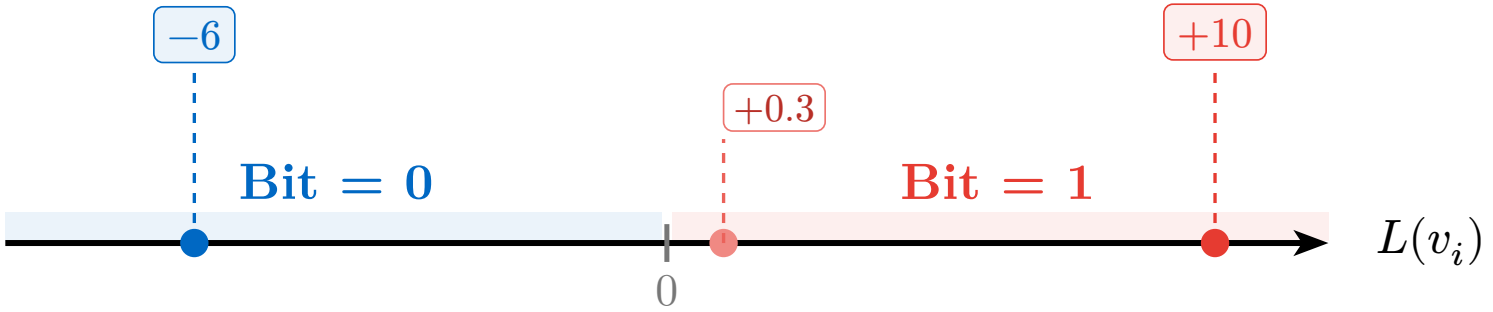
Décodage Soft : Le LLR

Signal

On reçoit une valeur y_i (ex: $+4.5$V ou -0.2V). Le LLR transforme cette mesure physique en une valeur statistique sans unité.

Log-Likelihood Ratio (LLR)

$$L(v_i) = \ln\left(\frac{P(v_i = 0 \mid y_i)}{P(v_i = 1 \mid y_i)}\right)$$



Signe
Définit la valeur du bit

|Valeur|
Confiance dans la décision

Sum-Product : Belief Propagation

Décodage Optimal

Échange itératif de croyances (LLR) entre les nœuds du graphe

Information Extrinsèque

Exclure l'avis du destinataire pour éviter l'auto-influence

Mise à jour CN

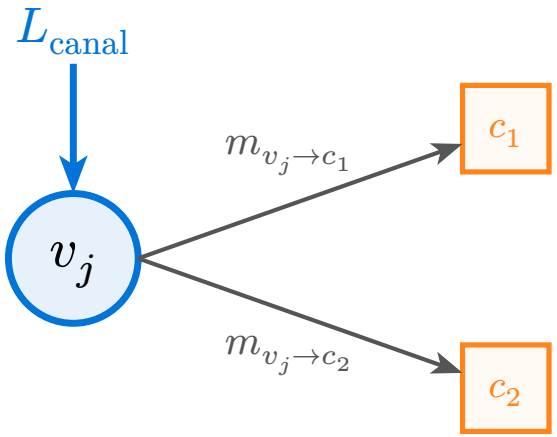
$$\tanh\left(\frac{m_{c\to v}}{2}\right) = \prod_{u\neq v} \tanh\left(\frac{m_{u\to c}}{2}\right)$$

Mise à jour VN

$$m_{v\to c} = L_{\text{canal}} + \sum_{c'\neq c} m_{c'\to v}$$

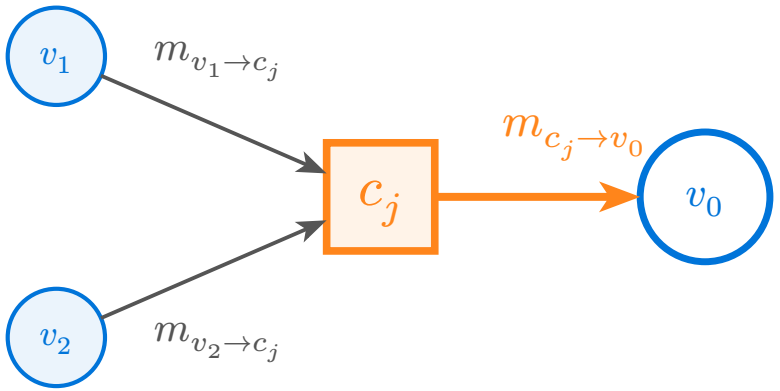
Sum-Product

Initialisation



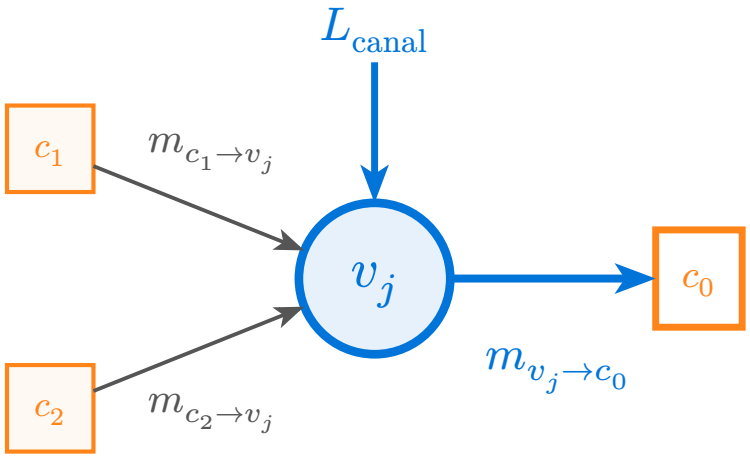
$$m_{v_j \to c_i} = L_{\text{canal}}$$

Échange CN

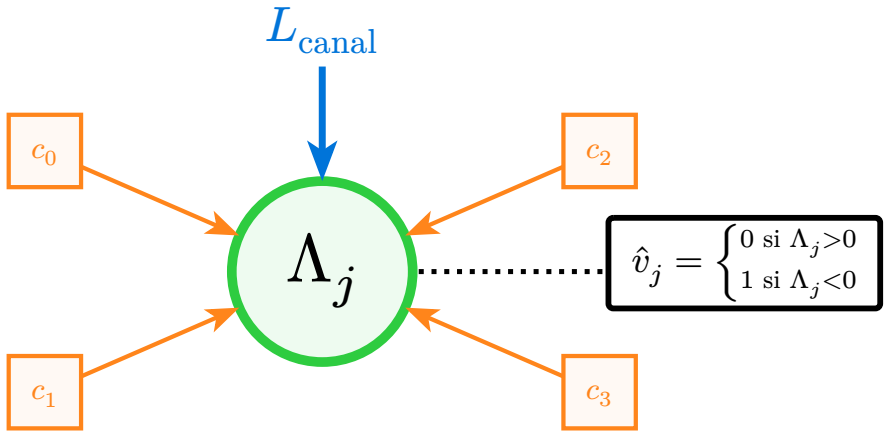


Itérations
$i = 1, ..., I_{\max}$

Échange VN



Décision Finale



$$\Lambda_j = L_{\text{canal}} + \sum_{i \in \mathcal{N}(v_j)} m_{c_i \to v_j}$$

Implementation rust

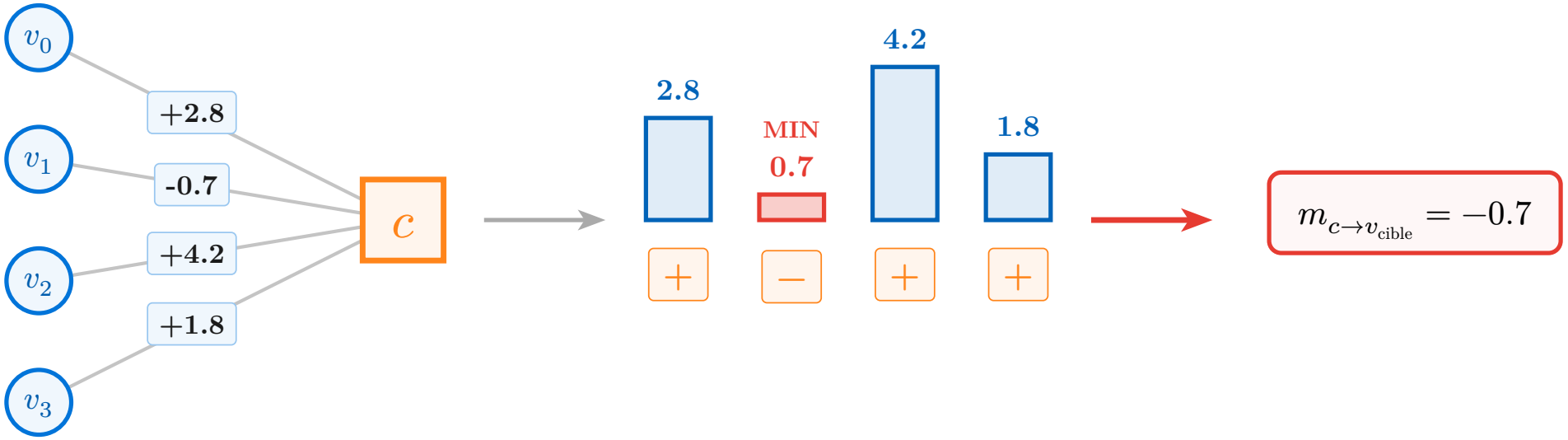
Min-Sum

Avantage Matériel

- **Comparateurs** pour le minimum
- **XOR** pour le produit des signes

Mise à jour du CN

$$m_{c \to v_i} = \prod_{j \neq i} \text{sgn}\left(m_{v_j \to c}\right) \times \min_{j \neq i} |m_{v_j \to c}|$$



Pour les VN : $m_{v \to c} = L_{\text{canal}} + \sum_{c' \neq c} m_{c' \to v}$

Test réel

Irl hackrf, test de diff de debit avec des paquets

Image

Test de transmission d'image avec différent ldpc non opti et opti (le H)

Annexe

Théorie deriere la définition des codes linaires

Poser les notations algebriques etc...

Decodage par maximum de vraisemblance

Expliquer, quelle distance ? etc

Code LDPC non régulier

Maths deriere Belief Propagation