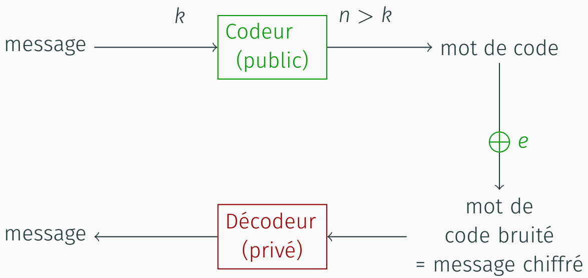
Cryptographie post-quantique : étude du décodage des codes QC-MDPC

Stage effectué à Inria Paris sous la supervision de Nicolas Sendrier

Valentin Vasseur

Septembre 2017

Chiffrement à clé publique de McEliece [McE78]



Difficulté : trouver des codes correcteurs sûrs et efficaces

Une variante du chiffrement de McEliece basée sur les MDPC [MTSB13]

Paramètres : $n_0, p, d, t \in \mathbb{N}$, $n = n_0 p$, p premier, d impair, $n_0 d \sim t \sim \sqrt{n}$

$H \leftarrow \mathbb{F}_2^{p \times n}$
Poids des lignes de H : $n_0 d$
$G = (I_p|\tilde{G}) \in \mathbb{F}_2^{(n-p)\times n}$ matrice génératrice correspondant à H

$\xrightarrow{\quad G \quad}$

$m \in \{0,1\}^{n-p}$
$c = mG$
$e \leftarrow \{0,1\}^n$
$|e| = t$

$\xleftarrow{\quad y = c + e \quad}$

$m = \text{Decode}(y, H)$

Une variante du chiffrement de McEliece basée sur les QC-MDPC [MTSB13]

QC-MDPC : Quasi-Cyclic Moderate Density Parity Check

$$H = \left(\begin{array}{|c|c|c|} H_0 & \cdots & H_{n_0} \end{array}\right)$$

- Clés de faible taille contrairement au système d'origine
- Preuves de sécurité
- Décodeur efficace

Algorithme de décodage (*bit-flipping*)

Propriété : Si H est une matrice de parité de $\mathcal{C}$ alors

$$y \in \mathcal{C} \iff Hy^\top = 0\,.$$

procedure $\textsc{Bit-flipping}(y, H)$ $\triangleright\ y = (y_1, \ldots, y_n) \in \{0,1\}^n$

  while $Hy^\top \neq 0$ **do** $\triangleright\ H = (h_1, \ldots, h_n) \in \{0,1\}^{p \times n}$

    $s \leftarrow Hy^\top$

    for $j = 1, \ldots, n$ **do**

      if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \text{seuil}$ **then**

        $y_j \leftarrow 1 - y_j$

  return y

Propriété : Si H est « suffisamment creuse » et y « suffisamment proche » de $\mathcal{C}$ alors l'algorithme converge.

Pour les QC-MDPC :

- Poids des lignes $O(\sqrt{n})$
- Correction de $O(\sqrt{n})$ erreurs

Exemple de décodage (*bit-flipping*)

procedure BIT-FLIPPING(y, H)
  $\mathbf{y} \leftarrow y$
  while $H\mathbf{y}^\top \neq 0$ **do**
    $s \leftarrow H\mathbf{y}^\top$
    for $j = 1, \ldots, n$ **do**
      if $\sigma_j = \langle s, h_j \rangle \geq \mathrm{seuil}$ **then**
        $\mathbf{y}_j \leftarrow 1 - \mathbf{y}_j$
  return $\mathbf{y}$

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

$$\mathbf{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Exemple de décodage (*bit-flipping*)

procedure BIT-FLIPPING(y, H)
  $\mathsf{y} \leftarrow y$
  while $H\mathsf{y}^\top \neq 0$ **do**
    $s \leftarrow H\mathsf{y}^\top$
    for $j = 1, \ldots, n$ **do**
      if $\sigma_j = \langle s, h_j \rangle \geq \mathrm{seuil}$ **then**
        $\mathsf{y}_j \leftarrow 1 - \mathsf{y}_j$
  return y

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

$$\mathsf{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Exemple de décodage (*bit-flipping*)

procedure BIT-FLIPPING(y, H)
  $\mathbf{y} \leftarrow y$
  while $H\mathbf{y}^\top \neq 0$ **do**
    $s \leftarrow H\mathbf{y}^\top$
    for $j = 1, \dots, n$ **do**
      if $\sigma_j = \langle s, h_j \rangle \geq \mathrm{seuil}$ **then**
        $\mathbf{y}_j \leftarrow 1 - \mathbf{y}_j$
  return $\mathbf{y}$

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

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

Exemple de décodage (*bit-flipping*)

procedure BIT-FLIPPING(y, H)
    $\mathbf{y} \leftarrow y$
    while $H\mathbf{y}^\top \neq 0$ **do**
        $s \leftarrow H\mathbf{y}^\top$
        for $j = 1, \ldots, n$ **do**
            if $\sigma_j = \langle s, h_j \rangle \geq \text{seuil}$ **then**
                $\mathbf{y}_j \leftarrow 1 - \mathbf{y}_j$
    return $\mathbf{y}$

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

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

État de l'art du décodage par *bit-flipping* des MDPC

- LDPC : seuil $\max_j \sigma_j$

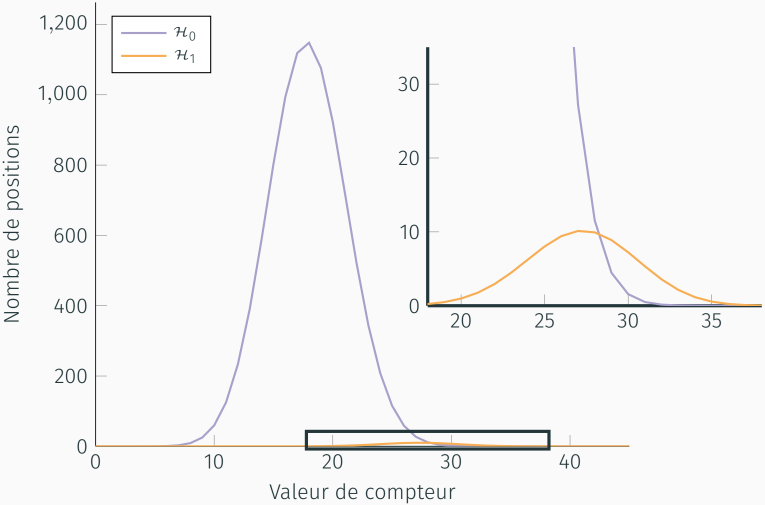
DFR : Decoding Failure Rate, Taux d'échec au décodage

État de l'art du décodage par *bit-flipping* des MDPC

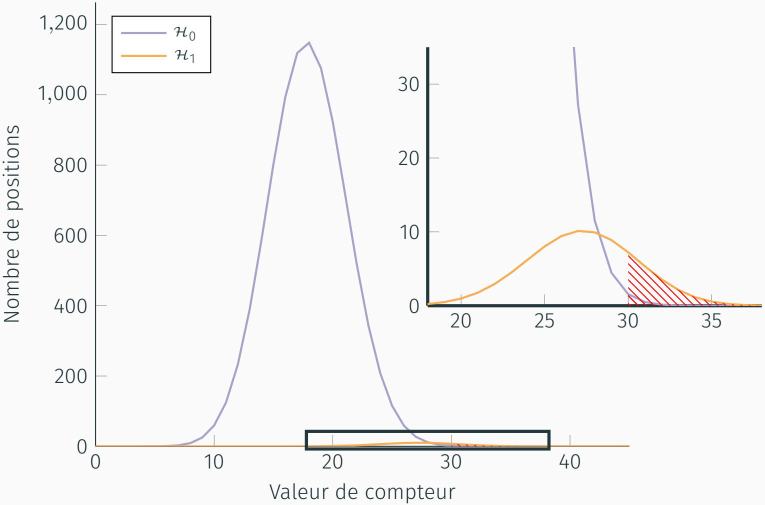
- LDPC : seuil $\mathsf{max}_j\, \sigma_j$
- [MTSB13] : seuil $\mathsf{max}_j\, \sigma_j - \Delta$ ($\Delta \approx 5$ fixé) $\mathrm{DFR} \approx 10^{-7}$

DFR : Decoding Failure Rate, Taux d'échec au décodage

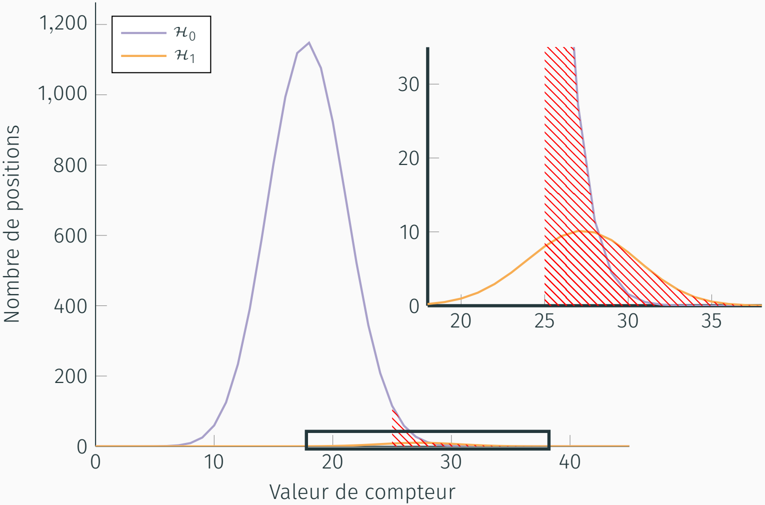
$\max_j \sigma_j - \Delta$



$\max_j \sigma_j - \Delta$



$\max_j \sigma_j - \Delta$

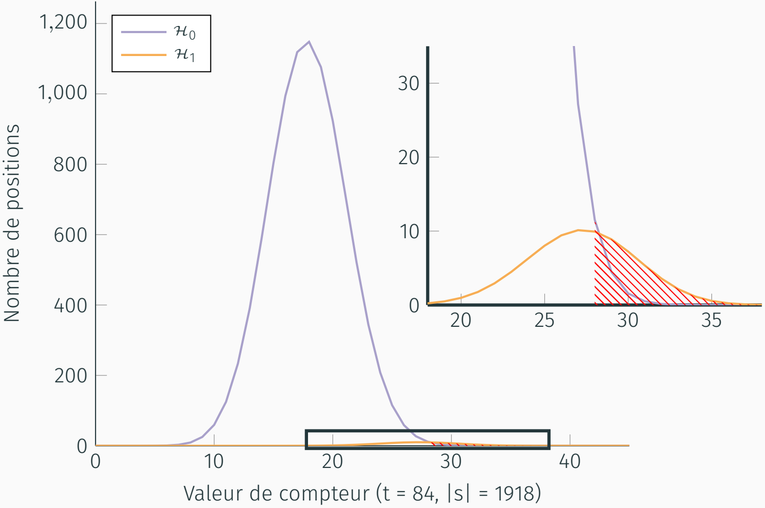


État de l'art du décodage par *bit-flipping* des MDPC

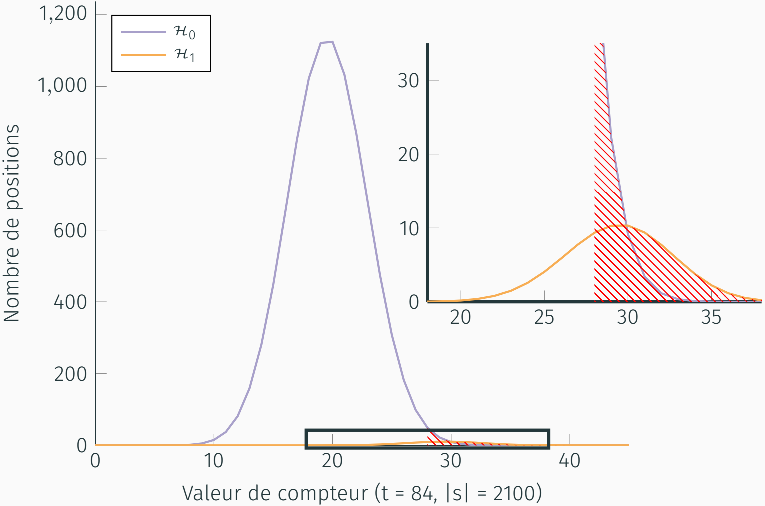
- LDPC : seuil $\max_j \sigma_j$
- [MTSB13] : seuil $\max_j \sigma_j - \Delta$ ($\Delta \approx 5$ fixé) $\mathrm{DFR} \approx 10^{-7}$
- [Cho16] : seuils fixes précalculés pour chaque itération $\mathrm{DFR} < 10^{-8}$

DFR : Decoding Failure Rate, Taux d'échec au décodage

Seuils fixes précalculés



Seuils fixes précalculés

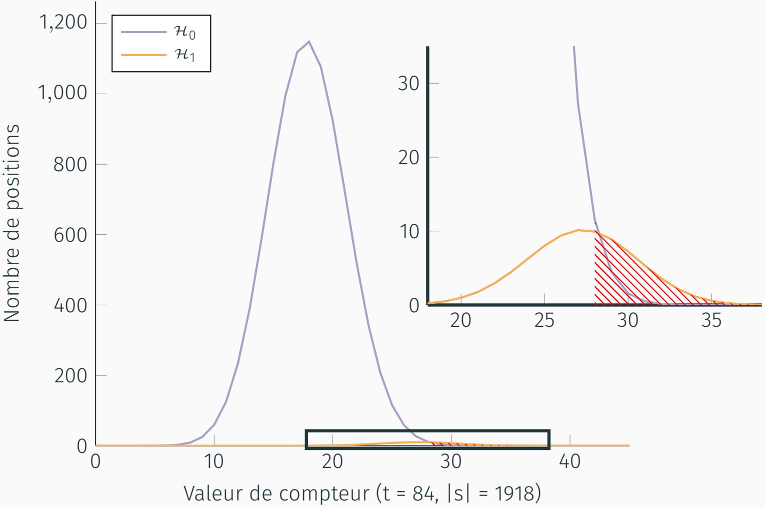


État de l'art du décodage par *bit-flipping* des MDPC

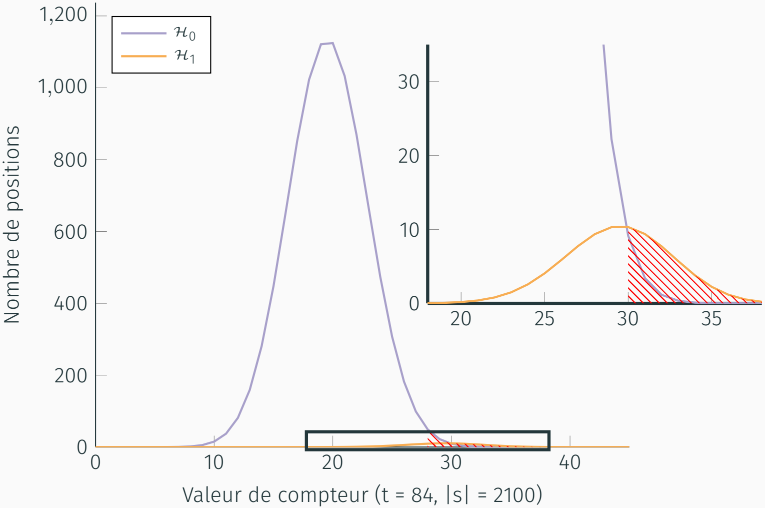
- LDPC : seuil $\max_j \sigma_j$
- [MTSB13] : seuil $\max_j \sigma_j - \Delta$ ($\Delta \approx 5$ fixé) $\mathrm{DFR} \approx 10^{-7}$
- [Cho16] : seuils fixes précalculés pour chaque itération $\mathrm{DFR} < 10^{-8}$
- [Cha17] : seuils adaptatifs en fonction de l'instance $\mathrm{DFR} \approx 10^{-9}$

DFR : Decoding Failure Rate, Taux d'échec au décodage

Seuils dépendants du poids du syndrome



Seuils dépendants du poids du syndrome



État de l'art du décodage par *bit-flipping* des MDPC

- LDPC : seuil $\max_j \sigma_j$
- [MTSB13] : seuil $\max_j \sigma_j - \Delta$ ($\Delta \approx 5$ fixé) $\quad \mathrm{DFR} \approx 10^{-7}$
- [Cho16] : seuils fixes précalculés pour chaque itération $\quad \mathrm{DFR} < 10^{-8}$
- [Cha17] : seuils adaptatifs en fonction de l'instance $\quad \mathrm{DFR} \approx 10^{-9}$

DFR : Decoding Failure Rate, Taux d'échec au décodage

Ingénierie :

- QC-MDPC candidats à être un standard de cryptographie post-quantique

$\rightarrow$ Comprendre le décodage pour l’implémenter

Sécurité :

- Une attaque exploite une corrélation entre les instances provoquant l’échec du décodeur et la clé secrète pour retrouver celle-ci entièrement [GJS16]

$\rightarrow$ Réduire le taux d’échec au décodage à une valeur très petite

Contributions pendant le stage

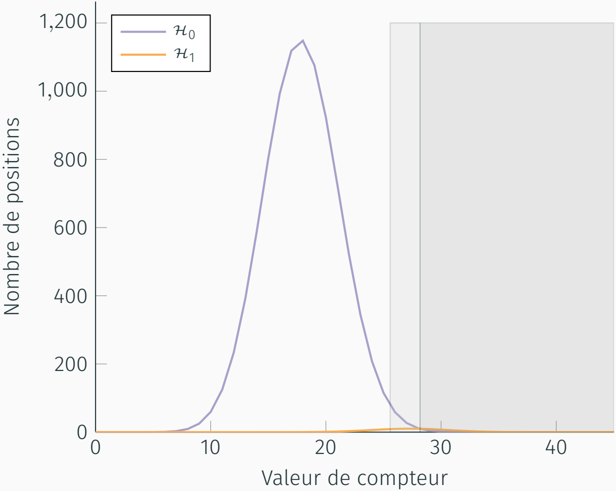
Décodage souple (état de l'art pour les codes LDPC) :

- Calcule le *rapport de vraisemblance* pour chaque position et chaque équation de parité que l'on affine à chaque itération
- Très coûteux en temps et mémoire

Intermédiaire :

- Ajout d'informations de fiabilité pour chaque position
- Éventuellement limiter les calculs aux positions les moins fiables
- Tout en restant peu coûteux en temps et mémoire

Zones grises



Zones grises

procedure Grey Bitflipping(y, H) $\quad\triangleright\ y = (y_1, \dots, y_n) \in \{0,1\}^n$

  $I \leftarrow 0$ $\quad\triangleright\ H = (h_1, \dots, h_n) \in \{0,1\}^{p \times n}$

  $G \leftarrow \emptyset$

  while $Hy^\intercal \neq 0$ **do**

    $s \leftarrow Hy^\intercal$

    if I is a multiple of I_G **then**

      $G \leftarrow \emptyset$

      for $j \in \{0, \dots, n\}$ **do**

        if $\sigma_j = \langle s, h_j^\top \rangle_\mathbb{Z} \geq \mathrm{seuil}_G$ **then**

          $G \leftarrow G \cup \{j\}$

    for $j \in G$ **do**

      if $\sigma_j = \langle s, h_j^\top \rangle_\mathbb{Z} \geq \mathrm{seuil}$ **then**

        $y_j \leftarrow 1 - y_j$

    $I \leftarrow I + 1$

  return y

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

$$\mathbf{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Zones grises

procedure Grey Bitflipping(y, H) ▷ $y = (y_1, \ldots, y_n) \in \{0,1\}^n$

  $I \leftarrow 0$ ▷ $H = (h_1, \ldots, h_n) \in \{0,1\}^{p \times n}$

  $G \leftarrow \emptyset$

  while $Hy^\intercal \neq 0$ **do**

    $s \leftarrow Hy^\intercal$

    if I is a multiple of I_G **then**

      $G \leftarrow \emptyset$

      for $j \in \{0, \ldots, n\}$ **do**

        if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \mathrm{seuil}_G$ **then**

          $G \leftarrow G \cup \{j\}$

    for $j \in G$ **do**

      if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \mathrm{seuil}$ **then**

        $y_j \leftarrow 1 - y_j$

    $I \leftarrow I + 1$

  return y

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

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

Zones grises

procedure GREY BITFLIPPING(y, H) $\triangleright\ y = (y_1, \ldots, y_n) \in \{0,1\}^n$

  $I \leftarrow 0$ $\triangleright\ H = (h_1, \ldots, h_n) \in \{0,1\}^{p \times n}$

  $G \leftarrow \emptyset$

  while $Hy^\intercal \neq 0$ **do**

    $s \leftarrow Hy^\intercal$

    if I is a multiple of I_G **then**

      $G \leftarrow \emptyset$

      for $j \in \{0, \ldots, n\}$ **do**

        if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \mathrm{seuil}_G$ **then**

          $G \leftarrow G \cup \{j\}$

    for $j \in G$ **do**

      if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \mathrm{seuil}$ **then**

        $y_j \leftarrow 1 - y_j$

    $I \leftarrow I + 1$

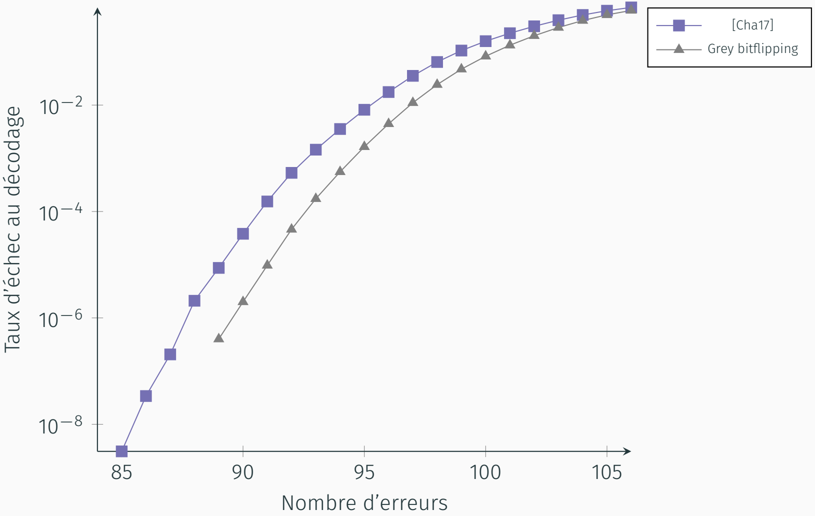
  return y

$$\sigma = \begin{pmatrix} 1 & 2 & 2 & 1 & 3 & 1 & 1 & 3 & 1 & 1 & 2 & 3 & 1 & 2 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

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

Décodage des QC-MDPC pour des poids d'erreur surdimensionnés



2-Bitflipping

procedure 2-BITFLIPPING(y, H) ▷ $y = (y_1, \dots, y_n) \in \{0,1\}^n$
    $z \leftarrow (0, \dots, 0) \in \{0,1\}^n$ ▷ $H = (h_1, \dots, h_n) \in \{0,1\}^{p \times n}$
    while $Hy^\intercal \neq 0$ **do**
        $s \leftarrow Hy^\intercal$
        for $j = 1, \dots, n$ **do**
            $\sigma_j \leftarrow \langle s, h_j^\top \rangle_{\mathbb{Z}}$
            $(y_j, z_j) \leftarrow f(y_j, z_j, \sigma_j)$
    return y

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

$$\mathbf{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$
\begin{array}{ll}
\textbf{procedure } \textsc{2-Bitflipping}(y, H) & \triangleright\ y = (y_1, \ldots, y_n) \in \{0,1\}^n \\
\quad z \leftarrow (0, \ldots, 0) \in \{0,1\}^n & \triangleright\ H = (h_1, \ldots, h_n) \in \{0,1\}^{p \times n} \\
\quad \textbf{while } Hy^\intercal \neq 0 \textbf{ do} & \\
\quad\quad s \leftarrow Hy^\intercal & \\
\quad\quad \textbf{for } j = 1, \ldots, n \textbf{ do} & \\
\quad\quad\quad \sigma_j \leftarrow \langle s, h_j^\top \rangle_{\mathbb{Z}} & \\
\quad\quad\quad (y_j, z_j) \leftarrow f(y_j, z_j, \sigma_j) & \\
\quad \textbf{return } y &
\end{array}
$$

$$
\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}
$$

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

$$
\mathrm{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}
$$

procedure 2-BITFLIPPING(y, H) $\qquad \triangleright\ y = (y_1, \ldots, y_n) \in \{0,1\}^n$

  $z \leftarrow (0, \ldots, 0) \in \{0,1\}^n$ $\qquad \triangleright\ H = (h_1, \ldots, h_n) \in \{0,1\}^{p \times n}$

  while $Hy^\intercal \neq 0$ **do**

    $s \leftarrow Hy^\intercal$

    for $j = 1, \ldots, n$ **do**

      $\sigma_j \leftarrow \langle s, h_j^\top \rangle_{\mathbb{Z}}$

      $(y_j, z_j) \leftarrow f(y_j, z_j, \sigma_j)$

  return y

$$\sigma = \begin{pmatrix} 1 & 2 & 2 & 1 & 3 & 1 & 1 & 3 & 1 & 0 & 2 & 3 & 1 & 2 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

$$\mathrm{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

procedure 2-BITFLIPPING(y, H) $\quad \triangleright\, y = (y_1, \dots, y_n) \in \{0,1\}^n$

$\quad z \leftarrow (0, \dots, 0) \in \{0,1\}^n$ $\quad \triangleright\, H = (h_1, \dots, h_n) \in \{0,1\}^{p \times n}$

$\quad$ **while** $Hy^\intercal \neq 0$ **do**

$\quad\quad s \leftarrow Hy^\intercal$

$\quad\quad$ **for** $j = 1, \dots, n$ **do**

$\quad\quad\quad \sigma_j \leftarrow \langle s, h_j^\top \rangle_{\mathbb{Z}}$

$\quad\quad\quad (y_j, z_j) \leftarrow f(y_j, z_j, \sigma_j)$

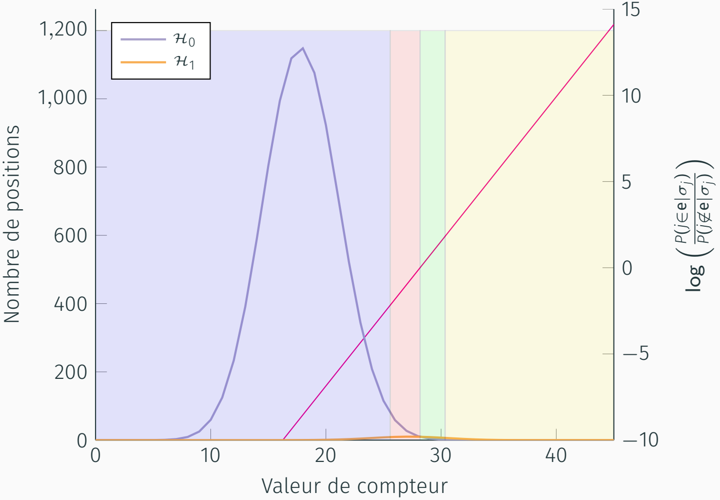
$\quad$ **return** y

$$\sigma = \begin{pmatrix} 1 & 2 & 2 & 1 & 3 & 1 & 1 & 3 & 1 & 0 & 2 & 3 & 1 & 2 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

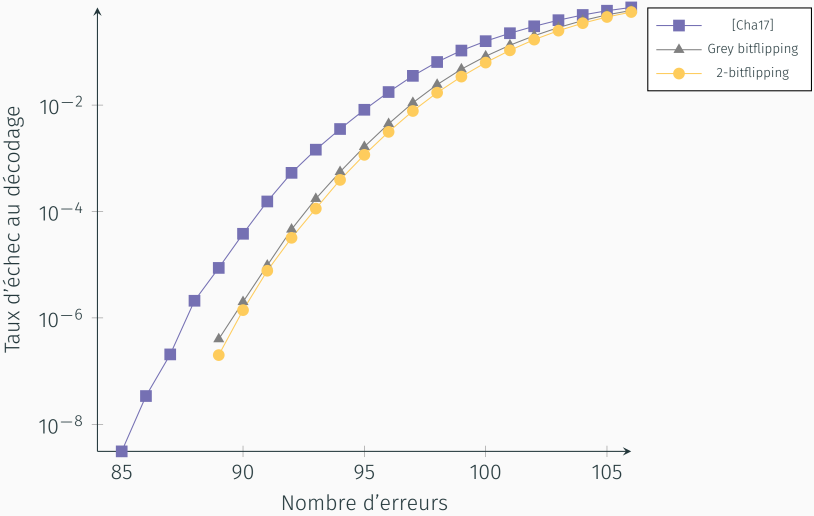
$$\mathrm{y} - y = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \end{pmatrix}$$

Définition de f

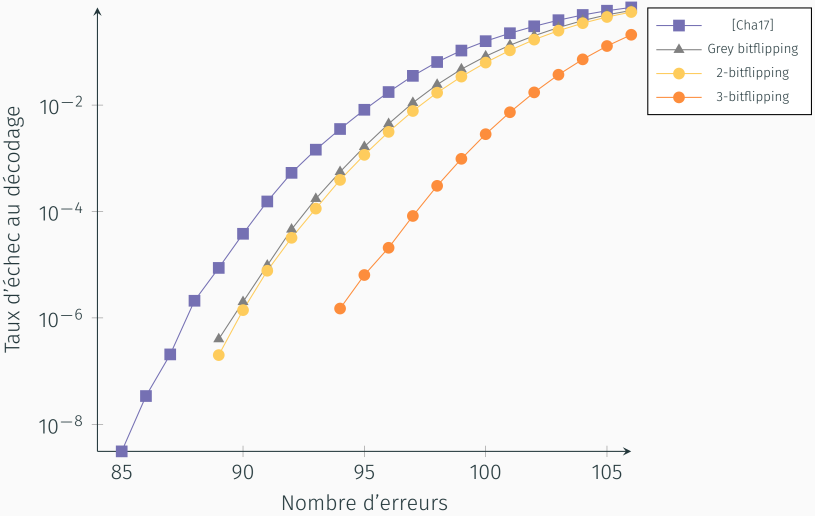


	Zone			
0	0	0	0	1
1	1	1	1	0
0	0	0	1	1
1	1	1	0	0

Décodage des QC-MDPC pour des poids d'erreur surdimensionnés



Décodage des QC-MDPC pour des poids d'erreur surdimensionnés



Décodage dynamique

procedure Dynamic Bitflipping(y, H) ▷ $y = (y_1, \ldots, y_n) \in \{0,1\}^n$
  while $Hy^\intercal \neq 0$ **do** ▷ $H = (h_1, \ldots, h_n) \in \{0,1\}^{p \times n}$
    $s \leftarrow Hy^\intercal$
    for $i = 1, \ldots, p$ **do**
      if $s_i \neq 0$ **then**
        for $j \in h_i$ **do**
          if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \mathrm{seuil}$ **then**
            $y_j \leftarrow 1 - y_j$
  return y

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

$$\mathrm{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Décodage dynamique

```
procedure DYNAMIC BITFLIPPING(y, H)            ▷ y = (y_1, ..., y_n) ∈ {0,1}^n
    while Hy^T ≠ 0 do                          ▷ H = (h_1, ..., h_n) ∈ {0,1}^{p×n}
        s ← Hy^T
        for i = 1, ..., p do
            if s_i ≠ 0 then
                for j ∈ h_i do
                    if σ_j = ⟨s, h_j^⊤⟩_ℤ ≥ seuil then
                        y_j ← 1 − y_j
    return y
```

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

$$\mathbf{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Décodage dynamique

procedure Dynamic Bitflipping(y, H) $\triangleright\ y = (y_1, \ldots, y_n) \in \{0,1\}^n$

  while $Hy^\intercal \neq 0$ **do** $\triangleright\ H = (h_1, \ldots, h_n) \in \{0,1\}^{p\times n}$

    $s \leftarrow Hy^\intercal$

    for $i = 1, \ldots, p$ **do**

      if $s_i \neq 0$ **then**

        for $j \in h_i$ **do**

          if $\sigma_j = \langle s, h_j^\top \rangle_{\mathbb{Z}} \geq \mathrm{seuil}$ **then**

            $y_j \leftarrow 1 - y_j$

  return y

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

$$\mathbf{y} - y = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

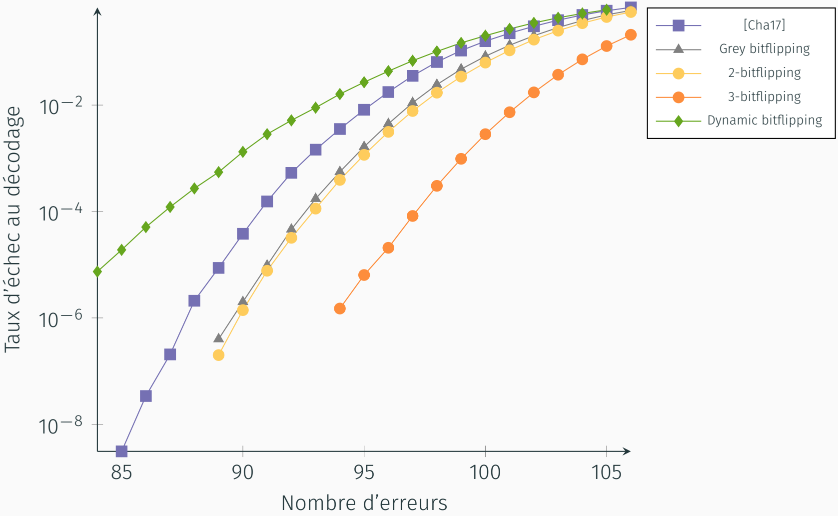
```
procedure DYNAMIC BITFLIPPING(y, H)          ▷ y = (y_1, ..., y_n) ∈ {0,1}^n
    while Hy^T ≠ 0 do                        ▷ H = (h_1, ..., h_n) ∈ {0,1}^{p×n}
        s ← Hy^T
        for i = 1, ..., p do
            if s_i ≠ 0 then
                for j ∈ h_i do
                    if σ_j = ⟨s, h_j^T⟩_Z ≥ seuil then
                        y_j ← 1 − y_j
    return y
```

$$\sigma = \begin{pmatrix} 2 & 2 & 2 & 2 & 3 & 2 & 1 & 3 & 1 & 1 & 2 & 3 & 2 & 3 & 2 & 1 & 2 & 2 \end{pmatrix}$$

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

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

Décodage des QC-MDPC pour des poids d'erreur surdimensionnés



Choix des seuils [Cha17]

Choix du seuil :

$$(n-t)\sum_{i\geq \text{seuil}} \binom{d}{i}\pi_0^i(1-\pi_0)^{d-i} < t\sum_{i\geq \text{seuil}} \binom{d}{i}\pi_1^i(1-\pi_1)^{d-i}$$

À la première itération :

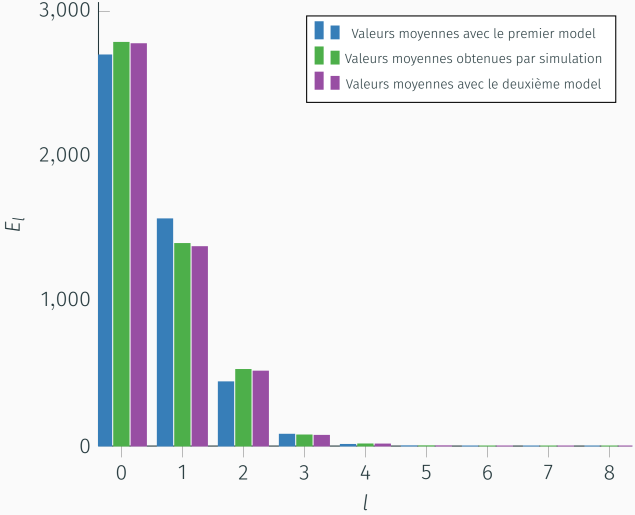
$$\pi_0 = \frac{1}{d(n-t)}\left((w-1)|s| - \sum_{\substack{l=0 \\ l \text{ odd}}}^{w}(l-1)E_l\right) ;$$

$$\pi_1 = \frac{1}{dt}\left(|s| + \sum_{\substack{l=0 \\ l \text{ odd}}}^{w}(l-1)E_l\right) .$$

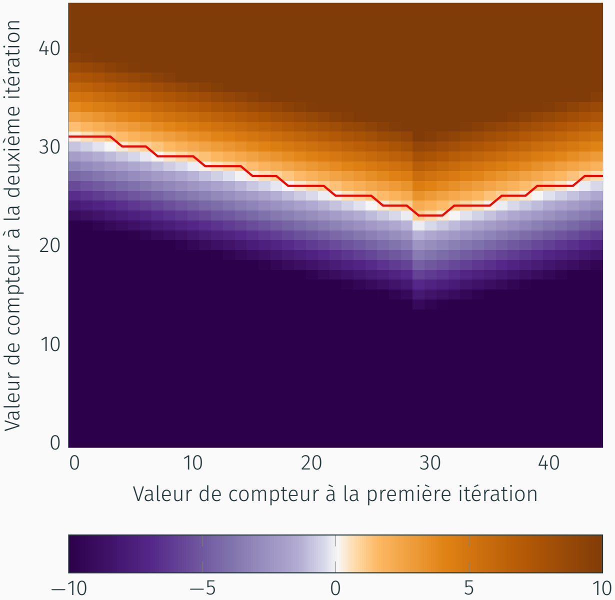
À partir de la deuxième itération :

Formules fausses

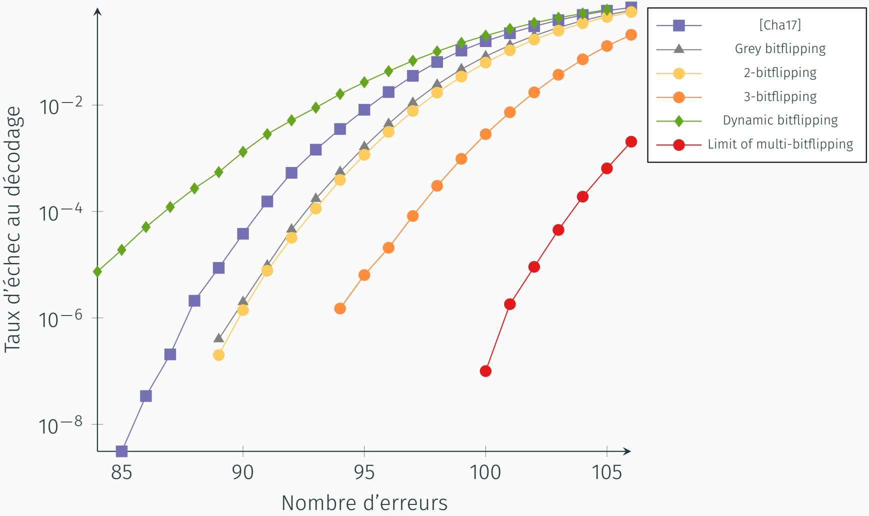
Distribution des E_l à la seconde itération



Évolution des compteurs entre la première et la deuxième itération



Décodage des QC-MDPC pour des poids d'erreur surdimensionnés



Conclusion

Variantes de l'algorithme de décodage

- Zones grises
  - → Réduit la compléxité
  - → Réduit le taux d'échec au décodage
- *b*-bitflipping
  - → Plus complexe
  - → Réduit grandement le taux d'échec au décodage
- Décodage dynamique
  - → Très simple
  - → Augmente grandement le taux d'échec au décodage

Évolution d'une itération à l'autre

- des valeurs de E_l
- des compteurs

Poursuite

- Amélioration des variantes connaissant les évolutions des E_l ou des compteurs
- Utilisation des points forts de chaque variante
- Estimer le taux d'échec au décodage de manière plus théorique

References

Julia Chaulet. “Étude de cryptosystèmes à clé publique basés sur les codes MDPC quasi-cycliques”. Thèse de doct. University Pierre et Marie Curie, 2017.

Tung Chou. “QcBits : Constant-Time Small-Key Code-Based Cryptography”. In : *Cryptographic Hardware and Embedded Systems - CHES 2016 - 18th International Conference, Santa Barbara, CA, USA, August 17-19, 2016, Proceedings*. 2016, p. 280–300. DOI : `10.1007/978-3-662-53140-2_14`. URL : `https://doi.org/10.1007/978-3-662-53140-2_14`.

Qian Guo, Thomas Johansson et Paul Stankovski. “A Key Recovery Attack on MDPC with CCA Security Using Decoding Errors”. In : *Advances in Cryptology - ASIACRYPT 2016 - 22nd International Conference on the Theory and Application of Cryptology and Information Security, Hanoi, Vietnam, December 4-8, 2016, Proceedings, Part I*. 2016, p. 789–815. DOI : `10.1007/978-3-662-53887-6_29`. URL : `http://dx.doi.org/10.1007/978-3-662-53887-6_29`.

Robert J McEliece. “A public-key cryptosystem based on algebraic”. In : *Coding Thv* 4244 (1978), p. 114–116.

Rafael Misoczki et al. “MDPC-McEliece : New McEliece variants from Moderate Density Parity-Check codes”. In : *Proceedings of the 2013 IEEE International Symposium on Information Theory, Istanbul, Turkey, July 7-12, 2013*. 2013, p. 2069–2073. DOI : `10.1109/ISIT.2013.6620590`. URL : `http://dx.doi.org/10.1109/ISIT.2013.6620590`