Th. Info. TD6. Corrigé

Titre de la note 01/04/2010

Ex 1

Partie 1. Expérience. Il suffit d'exécuter le fichier image.sci dans scilab. La probabilité d'erreur se règle à l'aide d'une petite fenêtre de dialogue qui s'ouvre.

La matrice de cumul des erreurs s'affiche à la fin sur la console. (penser à saisir "Enter" pour afficher toute la matrice)

(3) Lien entre la possibilité de correction et la matrice de cumul

Les mots du code sont $w_1 = 000$ et $w_2 = 111$

On décode "0" par vote majoritaire si après transmission il y a 2 ou 3 "0" dans le mot reçu. Idem pour "1".

Supposons qu'on envoie le mot "000"

- 0 erreurs $\Leftrightarrow$ on reçoit $m =$ "000". On décode "0".
- 1 erreur $\Leftrightarrow$

$$000 \to \left.\begin{array}{l} 100 \\ 010 \\ 001 \end{array}\right\}$$

on décode "0". Donc on corrige les erreurs.

- 2 erreurs

$$000 \to \left.\begin{array}{l} 101 \\ 110 \\ 011 \end{array}\right\}$$

on décode "1" (au lieu de "0"!) Donc 2 erreurs ne peuvent pas être corrigées

- 3 erreurs

000 → 111 → on décode "1"

Ici on ne se rend même pas compte qu'il y a eu des erreurs !

Conclusion le décodage est correct seulement s'il y a eu 0 ou 1 erreur sur les trois transmissions !

On peut observer que là où dans la matrice on a 2 ou 3 le pixel correspondant dans l'image décodée est inversé !

Partie 2 Analyse

1) Il s'agit de calculer la capacité de correction : $e_c = \frac{d-1}{2}$ où d est la distance min du code. Comme il n'y a que 2 mots $d = d_h(000, 111) = 3$.

$$\Rightarrow e_c = \frac{3-1}{2} = 1.$$

On confirme l'observation : on peut corriger maximum 1 erreur.

2) Ce code est en effet linéaire. $k=1, n=3$

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

En effet : $\begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \times 0 = \begin{pmatrix} 0 & 0 & 0 \end{pmatrix}$

$\begin{pmatrix} 1 & 1 & 1 \end{pmatrix} \times 1 = \begin{pmatrix} 1 & 1 & 1 \end{pmatrix}$

Ex 2

1) $k = 4$ (nb. de colonnes = taille de blocs d'origine)
$n = 8$ (nb de lignes = taille des mots de code)

2) Forme systématique $\begin{pmatrix} I_k \\ G' \end{pmatrix}$

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

On remplace la colonne C_4 par $C_4 + C_1 + C_2 + C_3$

$$\tilde{G} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} I_4 \\ G' \end{pmatrix}$$

3) On peut utiliser la matrice systématique.
En effet, les transformations sur les colonnes que nous avons utilisées font que G et $\tilde{G}$ sont deux matrices équivalentes $\Rightarrow$ elles ont la même image
$\Rightarrow$ elles engendrent le même ensemble de mots de code.

Pour calculer l'ensemble de mots de code on doit calculer

$$C = \{ Gm \, , \, m \in \{0,1\}^4 \}$$

Or, comme $G = \begin{pmatrix} I_4 \\ \hline G' \end{pmatrix}$ on a

$$Gm = \begin{pmatrix} m \\ G'm \end{pmatrix}$$ (on écrit les mots m en colonne: $\begin{pmatrix} m_1 \\ m_2 \\ m_3 \\ m_4 \end{pmatrix}$

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

On remarque que calculer Gm revient à faire toutes les sommes des trois sur 4 bits de m

$$G'm = \begin{pmatrix} m_1 + m_3 + m_4 \\ m_2 + m_3 + m_4 \\ m_1 + m_2 + m_4 \\ m_1 + m_2 + m_3 \end{pmatrix}$$

⚠ Les sommes sont modulo 2 !

m^T	$(Gm)^T = (m^T (G'm)^T)$	$\omega(w)$
0000	0000 0000	0
1000	1000 1011	4
0100	0100 0111 M_2	4
0010	0010 1101	4
0001	0001 1110	4
1100	1100 1100	4
1010	1010 0110	4
1001	1001 0101	4
0101	0101 1001	4
0011	0011 0011	4
0110	0110 1010	4
1110	1110 0001	4
1101	1101 0010	4
1011	1011 1000	4
0111	0111 0100	4
1111	1111 1111	8

4) Distance min d'un code linéaire = poids min non nul d'un mot de code. (voir propos. du cours). Donc il suffit de calculer les poids (nb. de bits à 1) voir ci-dessus.

On a : $d = 4. \Rightarrow e_c = E\left[\frac{d-1}{2}\right] = 1$

Max d'erreurs corrigées = 1

5) Matrice de contrôle

$$H = \left(G' \quad I_{8-4}\right)$$

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

6) Pour savoir si un mot reçu est un mot du code on utilise la propriété de la matrice de contrôle. On calcule le syndrome

$$S = Hm \quad \text{(toujours } m \text{ en colonne)}$$

et on a

$$m \in C \Leftrightarrow S = Hm = 0$$

(m est un mot du code $\Leftrightarrow S = 0$)

Soient $m_1 = 00010101$ $\quad m_3 = 00111010$

$m_2 = 01000111$

$$\overset{H}{\begin{pmatrix} 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}} \begin{array}{c} \begin{array}{ccc} m_1 & m_2 & m_3 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{pmatrix} \end{array} = \begin{array}{c} \begin{array}{ccc} S_1 & S_2 & S_3 \end{array} \\ \begin{pmatrix} 1 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 1 \end{pmatrix} \end{array}$$

$S_1 \neq 0 \Rightarrow m_1 \notin C$ $\quad S_2 = 0 \Rightarrow m_2 \in C$ (voir la table de mots pour vérifier)

$S_3 \neq 0 \Rightarrow m_3 \notin C$

7) Il y a au plus 1 erreur par mot.

On voit déjà que m_2 sera décodé par $\underbrace{0100}$ car c'est un mot de code.

D'ailleurs, s'il y a au plus une erreur alors m_2 est sans erreur. Car la distance min du code est 4. Cela veut dire qu'il faut 4 erreurs au minimum pour transformer un mot du code en un autre mot.

Considérons m_1. On sait déjà qu'il y a erreur car $S_1 \neq 0$. Soit e_1 le vecteur erreur.

$$m_1 = w_1 + e_1, \text{ avec } w_1 \in C$$

$\Rightarrow Hm_1 = He_1$ l'erreur a le même syndrome que m_1! Si on sait qu'il y a une seule erreur le vecteur e_1 s'est de la forme :

$$e_1 = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \text{-ième position.}$$

Alors $He_1 = C_i$ la ième colonne de H.

Donc s'il y a une seule erreur, au ième bit, on doit avoir :

$$C_i = He_1 = Hm_1 = S_1$$

l'une des colonnes de H doit être égale à $S_1 = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \end{pmatrix}$

On constate en effet que c'est la première colonne : $C_1 = S_1$.

Donc le vecteur erreur est $\begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}$

et on trouve le mot code

$$w_1 = m_1 + e_1 = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} + \begin{pmatrix} 1 \\ 0 \\ \vdots \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 1 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \end{pmatrix}$$

Considérons $m_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \end{pmatrix}$ $\quad S_3 = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}$

On constate que le syndrome n'est égal à aucune colonne de H. Donc il y a deux erreurs ou plus. Pour essayer de les trouver on teste les hypothèses de 2,3,4... erreurs.

Par exemple, si on suppose qu'il y a deux erreurs, c'est qu'il doit existez $i \neq j$ tq

$$C_i + C_j = H m_3 = \begin{pmatrix}1\\0\\0\\1\end{pmatrix}$$

Par exemple, $$C_5 + C_8 = \begin{pmatrix}1\\0\\0\\0\end{pmatrix} + \begin{pmatrix}0\\0\\1\\0\end{pmatrix} = \begin{pmatrix}1\\0\\0\\1\end{pmatrix}$$

On corrige donc avec $$e = \begin{pmatrix}0\\0\\0\\0\\1\\0\\0\\1\end{pmatrix}$$

$$W_3 = \begin{pmatrix}0\\0\\0\\1\\1\\1\\0\\0\end{pmatrix}$$

Attention! la capacité de correction ici est de 1. Cela veut dire que s'il y a 2 erreurs on peut ne pas pouvoir corriger.

Dans le cas de m_3 on peut remarquer que le vecteur de correction n'est pas unique car

$$C_5 + C_8 = C_1 + C_7 = C_3 + C_6 = S_3 !$$

Et cela donne 3 mots de code !