Travail pratique concernant une introduction au codage de canal

11 mars 2010

1 Introduction

Un code bloc $(n,\ k)$ est un code qui transforme des blocs de k bits d'information en blocs de n bits.

On travaillera dans la suite dans ce que l'on appelle un champ de Galois à 2 éléments 0 et 1, ce qui veut dire que toutes les opérations se font dans ce champ (on ne sort jamais du champ ni par addition, ni par multiplication).

Les règles que l'on utilisera par la suite sur les bits seront telles que

$$\begin{aligned} 0+0 &= 0 \\ 0+1 &= 1 \\ 1+0 &= 1 \\ 1+1 &= 0 \\ 0\times 0 &= 0 \\ 0\times 1 &= 0 \\ 1\times 0 &= 0 \\ 1\times 1 &= 1 \end{aligned}$$

Lorsqu'on aura à calculer la distance entre des mots de n bits, on utilisera une distance dite de Hamming telle que la distance entre les deux mots est le nombre de bits qui diffèrent position par position entre les deux mots, par exemple

$$d_{Hamming}([1,1,0,1,0,0],[0,0,0,1,0,0]) = 2$$

Un exemple de code primitif $(2,1)$ est donné ci-dessous

$$\begin{array}{ccc} 0 & \longrightarrow & 00 \\ 1 & \longrightarrow & 11 \end{array}$$

Pour ce code, les mots d'information sont 0, ou 1. Les mots-code sont 00 ou 11. Ce code primitif permet de créer de la redondance, au lieu d'envoyer un simple 0 sur le canal, on envoie 00. Dès lors si une erreur unique se produit par mot-code transmis, on recevra 01 ou 10. Ces 2 mots, 01 ou 10 ne sont pas des mots-code et dès lors on sait qu'on est en erreur. On voit donc que si une erreur unique a lieu par bloc on saura la détecter. Par contre si deux erreurs se produisent, on ne peut les détecter, car l'on passe d'un mot code admissible, par exemple 00 à un autre mot-code admissible 11 si deux erreurs se produisent. On dit que la *capacité de détection* du code est égal à 1. On peut définir la *capacité de détection* d'un code comme étant le nombre d'erreurs maximum qui seront systématiquement détectables pour un code donné. On peut facilement montrer que de façon générale, la capacité de détection d'un code est

$$Capa\ detection = d_{min} - 1$$

où d_{min} est la distance minimale qui existe entre les mots codes.

Un manière commode de générer des mots-code est d'utiliser une matrice dite génératrice telle que

$$y = x \times G$$

Par exemple la matrice génératrice du code primitif défini plus haut est

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

car

$$0 \times [1\ 1] = [0\ 0]$$

et

$$1 \times [1\ 1] = [1\ 1]$$

2 TP sur génération de codes blocs et calcul de la capacité de détection

Soit un code dont la matrice génératrice est

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

2.1 Partie pratique sur la capacité de détection

Ecrire un petit programme Matlab qui calcule

1. L'ensemble des mots d'information.
2. L'ensemble des mots-code.
3. La capacité de détection.
4. et qui code la séquence [000100111]

2.2 Partie théorique sur la capacité de détection

Puis

1. Vérifier d'abord à la main que la somme de deux mots-code est un mot-code.
2. Démontrez que la somme de deux mots-code est un mot code.
3. Démontrez que la différence de deux mots-code est un mot code.
4. Déduire de la question précédente que la capacité de détection est le nombre minimum de 1 dans le mot-code qui en compte le moins si l'on excepte le mot-code nul.

2.3 Réflexion sur le sens de la notion de mot-code

Créer un mot-code c'est créer une loi de dépendance entre des bits d'information et des bits de redondance.

Pour la matrice génératrice précédemment donnée, en désignant sous la forme $[x_1,\ x_2,\ x_3]$ les bits d'information, et sous la forme $[y_1,\ y_2,\ y_3, y_4,\ y_5,\ y_6,\ y_7]$ les bits de redondance,

1. Examinez où sont placés les bits d'information et les bits de redondance dans les mots-code.
2. Ecrire la loi mathématique qui relie les bits de redondance au bits de mots-code.
3. Dessiner sous forme de schéma contenant des ou exclusifs la dépendance entre bits d'information et bit de redondance.
4. Compléter le schéma pour faire un détecteur d'erreurs qui signale à la réception la présence d'une erreur.
5. Détectera-t-on toutes les erreurs ?

3 TP sur la notion de la capacité de correction

Lorsque la distance minimale de Hamming entre des mots-codes est suffisamment grande, on peut utiliser les codes-blocs pour corriger des erreurs.

1. En vous inspirant de la définition de la capacité de détection, définissez vous même une *Capacité de correction.*
2. Montrez que la capacité de correction s'écrit sous la forme

$$Capa\ correction = Partie\ Entiere\left(\frac{d_{min} - 1}{2}\right)$$

   où d_{min} est la distance minimale qui existe entre les mots codes.
3. Calculez la capacité de correction pour la matrice G définie plus haut.

4 TP sur la notion de syndrome

Dans la théorie des codes, un syndrome est une quantité qui permet de vérifier si un code est entaché d'une erreur ou pas.

1. Construire une matrice H orthogonale à G, c'est-à-dire vérifiant la propriété

$$G \times H' = 0$$

2. En déduire que pour un mot code C, on a forcément

$$C \times H' = 0$$

3. Déduire de la question précédente que pour un mot $\tilde{C}$ qui n'est pas mot code que

$$\tilde{C} \times H' \neq 0$$

4. En déduire la règle de calcul du syndrome.
5. Calculer le syndrome si le bloc reçu est [1111010] et conclure s'il y erreur ou pas.
6. Corriger à l'aide d'un petit programme que vous écrirez la séquence

$$0011111101111101111110011110$$

et retrouver la séquence d'information.