

TD1 – Détection et correction d'erreurs

1 Canal de communication

Exercice 1.1 : *Canal vraiment très bruité ?*

Supposons qu'un CBSSM soit vraiment très bruité, c'est-à-dire un canal dans lequel la probabilité p d'erreur lors de la transmission d'un bit soit strictement supérieure à $1/2$.

Q 1.1 Que dire de ce canal si $p = 1$? Est-il vraiment si bruité ?

Q 1.2 Que faire si $1/2 < p < 1$?

2 Détection/correction d'erreurs

Exercice 2.1 : *Numéro INSEE*

Le numéro INSEE est un numéro composé de 13 chiffres auquel on ajoute une clé de deux chiffres pour contrôler d'éventuelles erreurs de saisie.

On obtient cette clé

1. en réduisant le numéro N modulo 97 : $r = N \pmod{97}$;
2. en retranchant r à 97 : la clé est $k = 97 - r$.

Par exemple, pour le numéro INSEE $N = 2\,90\,02\,59\,350\,112$ on a

1. $r = N \pmod{97} = 96$;
2. $k = 97 - 96 = 01$.

La clé est un entier compris 1 et 97. On l'écrit en base 10 sur deux chiffres.

Q 1.1 Comment contrôler la validité d'un couple numéro/clé (N, k) ?

Q 1.2 Vérifier que l'ajout de cette clé au numéro INSEE permet de détecter toutes les erreurs simples (modification d'un chiffre).

Pour approfondir

Q 1.3 Vérifier que l'ajout de la clé permet de détecter les permutations de deux chiffres consécutifs.

Q 1.4 Permet-il de détecter davantage d'erreurs ?

Exercice 2.2 : *Capacité de correction*

Q 2.1 Calculez la capacité de détection et de correction des codes binaires suivants :

1. $C = \{11100, 01001, 10010, 00111\}$
2. $C = \{11111111, 11001100, 10101010, 00000000\}$.

Q 2.2 Soit d la capacité de détection d'erreurs d'un codage. Déduisez-en sa capacité de correction.

Exercice 2.3 : Codage de parité à deux dimensions

Voici une amélioration du codage de parité : on découpe le message de k bits à transmettre en k_1 blocs de k_2 bits, et on code séparément chacun de ces blocs. Pour cela

1. on dispose le bloc à coder en k_1 lignes de longueur k_2 ,
2. on ajoute un bit de parité à la fin de chacune de ces lignes,
3. on ajoute une $k_1 + 1$ -ème ligne en faisant le codage de parité des $k_2 + 1$ colonnes formées par les k_1 lignes qui précèdent,
4. puis on forme le mot de longueur $n = (k_1 + 1)(k_2 + 1)$ obtenu en concaténant ces $k_1 + 1$ lignes.

Par exemple, avec $k_1 = 5$ et $k_2 = 4$, pour transmettre le $\mathbf{u} = \overbrace{0010\ 1110\ 0101\ 0001\ 0110}^{k_2}$, on construit les lignes :

$$\begin{array}{c}
 \begin{array}{c} \overleftarrow{k_2} \rightarrow \\ \uparrow k_1 \downarrow \\ \begin{array}{l} 0010\ 1 \\ 1110\ 1 \\ 0101\ 0 \\ 0001\ 1 \\ 0110\ 0 \\ 1110\ 1 \end{array} \end{array}
 \end{array}$$

et on transmet le mot $\mathbf{c}(\mathbf{u}) = 0010\ 1110\ 0101\ 0001\ 0110\ 1101011101$ (bien entendu les espaces sont juste une aide visuelle mais ne sont pas transmises).

Q 3.1 Étudiez les propriétés de détection et correction d'erreurs de ce codage.

Q 3.2 Ce codage par parité à deux dimensions est un $[n, k, d]$ -codage linéaire. Donner les valeurs de n , k et d lorsque $k_1 = 2$ et $k_2 = 2$, puis de manière générale en fonction de k_1 et k_2 .

Q 3.3 Donner une matrice génératrice du codage linéaire pour $k_1 = 2$ et $k_2 = 2$.

Q 3.4 En déduire la transposée d'une matrice de contrôle.

Exercice 2.4 : Détection et correction d'erreurs

Q 4.1

1. Donner la liste des mots à une distance 1 de 0101.
2. Donner la liste des mots à une distance 2 de 0101.

Pour approfondir

Q 4.2 De manière générale, combien a-t-on de mots à une distance inférieure ou égale à d de mots de longueur n sur un alphabet de taille 2 ?

Nous souhaitons avoir un codage détecteur et correcteur d'erreur produisant des mots de longueur 8 sur un alphabet de taille 2 ($\mathbb{F}_2^8$).

Q 4.3 Un code composé de 6 mots de $\mathbb{F}_2^8$ peut-il être 3-correcteur ?

Q 4.4 Un code composé de 6 mots de $\mathbb{F}_2^8$ peut-il être 2-correcteur ?

Exercice 2.5 : Longueur minimale

On veut un codage binaire au moins 1-correcteur pour coder 16 mots.

Q 5.1 De combien de bits a-t-on besoin pour coder 16 mots, sans capacité de détection et correction ?

Q 5.2 Soit un mot v à distance 1 d'un mot du code, qui est 1-correcteur. De combien de mots du code, le mot v est-il à distance 1 ?

Q 5.3 Pour un mot u du code, combien existe-t-il au total de mots qui sont à distance au plus 1 de u ?

Q 5.4 Pour qu'un code de 16 mots soit 1-correcteur, combien de mots doivent exister ?

Q 5.5 Combien de mots de longueur n peut-on générer ?

Q 5.6 En déduire la valeur minimale de n pour générer un code 1-correcteur de 16 mots

Exercice 2.6 : Détection et correction d'erreurs

Q 6.1 Quelle est la distance minimale du code suivant ?

$L = \{0110101110, 0011011100, 1000101001, 1010100010\}$

Q 6.2 Donnez un code binaire, de trois mots de longueur 8 (appelés u , v et w), dont la distance minimale est 5.

Q 6.3 Combien d'erreurs un tel code permet-il de corriger ? Donnez un exemple où il est possible de corriger ce nombre maximal d'erreurs en expliquant pourquoi on peut effectivement corriger ces erreurs.

Q 6.4 Soit c la capacité de correction du code. Nous transmettons un mot v qui est transmis avec $c' > c$ erreurs, soit v' le mot reçu. Peut-on trouver un exemple pour lequel il est possible de corriger v' en v ?

Pour approfondir

Exercice 2.7 : Codage de Hamming

Le mot $\mathbf{v}' = 1100110$ a été reçu, en sachant que le mot émis a été codé avec le codage de Hamming $[7, 4, 3]$ (dont la matrice génératrice et une matrice de contrôle sont redonnées ci-dessous).

$$G = \left(\begin{array}{cccc|ccc} 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{array} \right) H = \left(\begin{array}{cccc|ccc} 1 & 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 \end{array} \right)$$

Q 7.1 En faisant l'hypothèse qu'au plus un bit est erroné, quel est le mot qui a été émis ? Quel est le message qui voulait être envoyé ?

Q 7.2 Combien y a-t-il eu d'erreurs lors de la transmission si le message initial était $u = 1100$? Pourquoi la correction n'a pas permis de retrouver le message d'origine ?