**SOLUTION TD2 (Partie 2) : CODAGE CANAL**

**CODAGE ET COMPRESSION MASTER 1 RT ET ST**

**Rappels de cours**

* Les codes linéaires, définies habituellement par la matrice génératrice G de taille k×n (où n est la taille d’un mot de code et k la taille d’un bloc de données original), sont des codes dont chaque mot du code CD est obtenu après transformation linéaire des bits du mot initial (message).
* Un codeur de bloc linéaire est souvent une transformation linéaire **Fk2 →Fn2**



* Pour le codage en bloc, notamment le codage linéaire, il s’agit de générer à partir des bits du message (par exemple de taille k) des bits de code (de taille n) formés des bits de message et des bits de contrôle ou de parité (de taille n-k).
* **F** est le corps fini.
* **F**2 est le plus petit corps fini composé uniquement de deux éléments 0 et 1.
* (F2)n est une fonction de n éléments dans corps **F**2
* Un code de longueur n est une partie de Fnq. Un code linéaire C de longueur n sur le corps fini Fq est un sous-espace vectoriel de Fnq. Par défaut, un code sera supposé linéaire.
* un **sous-espace vectoriel** d'un espace vectoriel Fnq*,* est une partie non vide *CD*, de Fnq, stable par combinaisons linéaires. Cette stabilité s'exprime par :
* ***la somme de deux vecteurs de CD appartient à CD ;***
* ***le produit d'un vecteur de CD par un scalaire appartient à CD.***
* La distance de Hamming entre deux vecteurs de Fnq est le nombre de coordonnées où les deux vecteurs diffèrent (ou les deux mots de code). La distance minimale dmin du code CD est la plus petite distance non nulle entre deux mots du code CD. dmin est aussi le plus petit poids wmin (nombre de coordonnées non nulles) d'un mot non nul de CD.
* Les paramètres [n, k, dmin] d'un code C sont sa longueur, sa dimension (en tant que Fq-espace vectoriel), sa distance minimale. Toute configuration de t < d/2 erreurs peut être corrigée en cherchant le mot de code le plus proche (pour la distance de Hamming).
* Une matrice génératrice d'un code C est une matrice k×n à éléments dans Fq, dont les lignes constituent une base de CD.
* Une matrice de contrôle ou de parité de CD est une matrice génératrice du code dual C⊥.
* Une matrice génératrice G est dite sous forme systématique si elle s'écrit:

G= [Ik|P]. où Ik est la matrice identité de taille k×k et P la matrice contenant les bits de Parité de taille k× (n-k)

* Dans ce cas on constate que la matrice H= [Pt|In−k] est une matrice de contrôle de C.
* Le syndrome de y’∈ Fnq est le vecteur (colonne)

s(Y) =HtY’.

Ou bien s(Y’) =∑y’ihi

Où Y’= (y’1,…y’n) et

h1…hn sont les colonnes de la matrice H.

* Si Y est un mot du code C (code original) de matrice de parité H si et seulement si s(Y) = 0
* Décodage par syndrome. Pour trouver le plus proche mot de code de Y, calculer s(Y), puis chercher le plus petit ensemble I⊂{1,2,…,n} tel qu'il existe des λi∈Fq non nuls, i∈I, et

∑λihi=s.

Exercice 1:

Soit un alphabet A={0, 1}, on définit un code tel que :

C={x1=(0,0,0,0), x2=(1,0,1,1), x3=(0,1,0,1), x4=(1,1,1,0)}

* 1. Est-il linéaire ?
	2. Quelle est la longueur de ce code ?
	3. Quel est son poids minimal ?
	4. Quelle est sa distance minimale ?
	5. Ce code, combien peut il corriger d’erreurs?

**Solution Exercice 1**

C est un code linéaire. Tout mot de code de C peut être obtenu par une somme de deux autres mots du même code

1. Oui, car la somme de deux de ses mots de code donne toujours un mot de ce même code

x2+x3=x4, x2+x4=x3, x3+x4=x2

1. La longueur du code n=4. Car chaque mot de code est formé par 4bits
2. Son poids minimal (où nombre d’éléments des mots de code non nuls le moins élevé) wmin=2
3. En calculant toutes les distances d(xi,xj) = {3, 2, 3, 3, 2, 3}, d’où la distance minimale dmin=2
4. Le nombre d’erreurs qu’il peut corriger est donné par **e=(dmin – 1)/2=0**. Par contre la capacité de détection d’erreurs d’un code est donnée par **e= dmin – 1=1**

Ce code peut détecter une erreur à la fois mais il ne peut pas corriger.

Exercice 2 :

Parmi ces codes binaires, lesquels sont linéaires ? Pour les codes linéaires déterminez (n, k, d) et quel est le nombre d’erreurs que chacun peut détecter ?

C1 = {00, 01, 11} ⊂ (F2)2 (1)

C2 = {000, 100, 110, 101, 011, 111} ⊂ (F2)3 (2)

C3 = {0000, 1100, 1010, 1001, 0110, 0101, 0011, 1111} ⊂ (F2)4 (3)

C4 = {10000, 01010, 00001} ⊂ (F2)5 (4)

C5 = {000000, 101010, 010101} ⊂ (F2)6 . (5)

C6 = {00000, 01010, 00001, 01011, 01001} ⊂ (F2)5  (6)

C7={000000,101000,001110,100111} ⊂ (F2)6 (7)

Solution Exercice 2 :

*Rappel : Pour qu’un code soit linéaire il faut que la somme de deux de ses mots de code donne toujours un mot de son code.*

* C1 n’est pas un code linéaire car il manque le mot de code 01 + 11 = 10
* C2 n’est par un code linéaire, car il lui manque 001, 110,
* C3 est un code linéaire, n’importe quel le somme de deux mots de code de C3 est un mot de code C3
* C4 n’est pas un code linéaire car il lui manque le mot de code 00000, mais aussi 11010, 01011, 100001
* C5, C6 et C7 ne sont pas des codes linéaires

**Exercice 3:**

Soit un code linéaire sur F2 dont la matrice génératrice est donnée ci-dessous



1. Trouvez la longueur des blocs du message (source), la longueur des mots de code et le nombre de bits de parité

2. Donnez la forme systématique de la matrice génératrice

3. En déduire la matrice de contrôle H

4. trouvez les mots de code

5. En déduire la distance minimale, le nombre d’erreurs que ce code peut détecter et peut corriger.

**Solution Exercice 3**

Rappel :

Une matrice génératrice d’un code linéaire CD sur le corps Κ est une matrice sur Κ dont les lignes forment une base de CD. C’est une matrice de k lignes et n colonnes.

Propriétés: Soit CD un code linéaire sur un corpsΚ.

Toute matrice génératrice est une matrice k × n (k la longueur des mots de la source et n celle des mots de code) sur Κ, avec k ≤ n, dont le rang est k;

Inversement, toute matrice k × n sur Κ de rang k, est une matrice génératrice d’un code (n,k) sur Κ



La matrice génératrice prend généralement la forme suivante :

Où [A] est une matrice de taille k×k et [P] est une matrice de taille **k × (n-k)**

**Si la matrice génératrice est sous sa forme normale ou systématique elle prend la forme suivante :**

Où [I] est une matrice identité de taille k×k et

1)- Dans notre cas k=3, n=6 et les bits de parité (redondance du code) sont au nombre de 3

2)- Rappel de cours :

**Théorème :**

***Deux k×n matrices G1 et G2 engendrent des (n,k)-codes linéaires équivalents si on peut obtenir G1 à partir G2 par une suite d'opérations à choisir parmi:***

* ***permutation des lignes ;***
* ***addition de deux lignes ;***
* ***permutation des colonnes.***

***Ce théorème peut être utilisé pour rendre une matrice génératrice quelconque sous forme normale ou systématique ou bien pour passer d’une matrice génératrice à une autre pour un même code linéaire.***

 Pour obtenir la matrice génératrice systématique nous pouvons entre autres réaliser des opérations d’additions (ou soustractions) entre des colonnes ou entre des lignes de cette matrice génératrice originale. Par exemple si on remplace la première ligne de la matrice G par la somme de la première et la deuxième ligne on obtient



Ensuite si on remplace la troisième colonne par la somme de la deuxième et troisième colonnes on obtient



Enfin, si on remplace la première colonne par la somme de la première et la troisième colonne on obtient :



La nouvelle matrice génératrice est sous forme normale ou systématique car elle prend la forme :

3)- Pour trouver la matrice de contrôle H, il suffit de rappeler que :

**H** est matrice de contrôle de parité du code linéaire permettant de définir les **n-k** bits de contrôle en fonction des bits d’information et du transposé du vecteur **CD** formé des bits codés.

⇒

Donc nous aurons



4)- Les mots de code CD sont obtenus à partir de la matrice génératrice G et la donnée D



Comme les données ont une taille k=3 (voir première question) alors nous avons 8 possibilités de données. Chacun de ces 8 vecteurs de données sera multiplié par les six colonnes de la matrice G pour trouver un mot de code d’une taille n=6.

|  |  |
| --- | --- |
| Donnée | Mot de code |
| 000 | 000000 |
| 001 | 001001 |
| 010 | 010101 |
| 011 | 011100 |
| 100 | 100111 |
| 101 | 101110 |
| 110 | 110010 |
| 111 | 111011 |

5)- dmin=2 ⇒ ce code peut détecter une seule erreur (e=dmin-1) sans pouvoir la corriger.

Exercice 4

Soit un code linéaire sur F2 dont la matrice génératrice est donnée ci-dessous

1. Trouvez la longueur des blocs du message (source), la longueur des mots de code et le nombre de bits de parité

2. En déduire la matrice de contrôle H

3. trouvez les mots de code

4. En déduire la distance minimale, le nombre d’erreurs que ce code peut détecter et peut corriger.

Solution Exercice 4

* + 1. On voit bien que la matrice génératrice G est déjà sous forme systématique (sous sa forme normale), c'est-à-dire composée de deux partie une première partie est une matrice identité de taille 2×2 donc k=2 (la taille du bloc de donnée du message initial) et la deuxième partie représente les bits de parité de taille 2×3 correspondant à k×(n-k) ce qui nous donne n=5 la taille d’un mot de code. **Il s’agit donc d’un code en bloc linéaire (5,2).**
		2. La matrice de contrôle de parité H peut être déduite à partir de la matrice génératrice

⇒

⇒

* + 1. Comme les blocs du message original ont une taille k=2, nous aurons alors 4 possibilités soit D={(0,0), (0,1), (1,0) et (1,1)} (où D désigne la donnée) pour chacun de ces blocs messages le codeur nous donnera un mot de code CD. Il suffit de faire le produit : **CD=DG**

|  |  |
| --- | --- |
| D : Donnée | CD : Mot de code |
| 00 | 00000 |
| 01 | 01111 |
| 10 | 10101 |
| 11 | 11010 |

4)- dmin=3 ⇒ ce code peut détecter deux erreurs (e=dmin-1) et il peut corriger une seule erreur.

Exercice 5

Un code linéaire a pour matrice de contrôle 

1. Préciser la longueur n des mots de code et la longueur k des mots d'information.
2. En déduire la matrice génératrice normale de ce code
3. Les données suivantes sont-elles des mots du code ?
	* C1 = (1 1 1 0 1 1)
	* C2 = (1 0 0 1 1 0)
4. Donner la matrice génératrice du code et le codage de chaque mot d'information.

**Solution exercice 5**

**Rappel : H est matrice de contrôle de parité du code linéaire permettant de définir les n-k bits de contrôle en fonction des bits d’information et du transposé du vecteur CD formé des bits codés.**

 **⇒**

**n c’est la taille du code, k la taille de la donnée du message (source) et n-k c’est la taille des bits de contrôle qu’on a rajouté dans le code**

a)- Dans notre cas et selon la matrice de contrôle proposée, on déduit que n-k=3, correspondant à la matrice identité, n=6 et donc k=3

b)-



**c)- Rappel : un mot codé CD  a été généré par la matrice génératrice *G* si et seulement si *HCDt* = 0.**



* Pour premier mot de code C1 = (1 1 1 0 1 1), , nous aurons

HC1T=010, le résultat n’est pas nul donc C1 n’est pas un mot de code

* Pour le deuxième mot de code C2 = (1 0 0 1 1 0), nous aurons

HC1T=000, donc C2 est un mot de code

**Exercice 6**

Lors d'un transfert de données, vous recevez les messages suivants codés grâce au code Hamming (7,4). Des erreurs s'y sont insérées. Retrouvez-les et corrigez-les.

* 0101000
* 1110010
* 1100011
* 1011011
* 1101011
* 1000011

**Solution exercice 6**

• 0101000 => 1101000

• 1110010 => 0110010

• 1100011 =>1101011

• 1011011 => 1011010

• 1101011 => 0101011

• 1000011 => 1000011

**Exercice 7**

On considère un code de Hamming (7,4).

1. Coder le message suivant : 010110010111
2. Décoder le message suivant : 010001110010101101001

**Solution exercice 7**

a. Codage de 0101 1001 0111 => 0101101 1001100 0111000

b. Décodage de: 0100011 1001010 1101001 => 0110011 0001010 1101000

Exercice 8

Soit Cun code polynomial obtenu par codage systématique, de générateur :

g(x) = x3+x2+x+1

1. Donner la longueur de la clé de contrôle des mots du code
2. Donner la matrice génératrice normalisée G5,2 du code C5,2de générateur g(x).
3. Donner les matrices génératrices des codes C6,3etC7,4ayant le même générateur g(x).

Exercice 9

Soit g(x) = x3+x+1 le polynôme générateur d'un code polynomial de longueur 6.

1. Quelle est la longueur des mots d'information ?
2. Evaluer le pourcentage de messages erronés reconnus comme tels parmi tous les messages erronés pour des erreurs par bit de probabilité p = 0,1.

Exercice 10:

Soit le codeur convolutif ci-dessous :



1. S’agit il d’un codeur RSC ou NRC ?
2. Quel est le rendement de ce codeur ?
3. Déterminez le mot de code de sortie en utilisant la transformée D pour la séquence d'entrée x(n) = (1001).
4. Dessinez le diagramme en treillis de ce codeur.
5. Trouvez la fonction de transfert et la distance minimale de ce codeur.

Exercice 11:

Soit le codeur convolutif ci-dessous :



1. S’agit il d’un codeur RSC ou NRC ?
2. Quel est le rendement de ce codeur ?
3. Trouvez la réponse impulsionnelle.
4. Trouvez la matrice des fonctions de transfert.
5. Utilisez la matrice de fonction de transfert pour déterminer le mot de code associé à la séquence d'entrée x = (11,10,01).

Exercice 12:

Considérons un codeur convolutif LTE (***Long Term Evolution*** est une évolution des normes de téléphonie mobile. En octobre 2010, l'UIT a reconnu la technologie LTE-Advanced , une évolution de LTE, comme une technologie 4G)de débit 1/3 avec une longueur de contrainte K = 7, comme le montre la figure suivante :



1. Trouvez la réponse impulsionnelle.
2. Trouvez la matrice des fonctions de transfert.

Exercice 13:

Considérons un codeur convolutif appelé ‘’tail-biting’’ du standard 802.16e (Réseaux informatiques sans fil WiMax) (figure ci-dessous). A noter que les codes convolutifs ‘’tail-biting’’ (TBCC) trouvent des applications dans de nombreuses normes de communication modernes telles que LTE et IEEE 802.16e.



1. S’agit il d’un codeur RSC ou NRC ?
2. Quel est le rendement de ce codeur ?
3. Trouvez la réponse impulsionnelle.
4. Trouvez la matrice des fonctions de transfert.

Exercice 14:

Soit le codeur suivant



* 1. S’agit il d’un codeur RSC ou NRC ?
	2. Trouvez les séquences génératrices de ce codeur convolutif
	3. Tracez la représentation du diagramme d'état de ce codeur

Exercice 15:

Soit le codeur suivant



* 1. S’agit il d’un codeur RSC ou NRC ?
	2. Trouvez les séquences génératrices de ce codeur convolutif
	3. Tracez la représentation du diagramme d'état de ce codeur