RLI : Projet 2021/2022 Spitz Christopher
Larbi Fariza

Rapport

1. Cette matrice est systématique car elle possède 4 colonnes qui représentent une matrice identité de taille 4.
Elle encode 4 vits de données.
La taille des mots du code est de 16.
Pour le rendement on a un code de longueur 16, 4 bits de données et 12
Après quelques recherches on trouve que le rendement d'un code de Hadamard est de k/2k. Ici k =4 dans notre cas car le code de Hadamard est de la forme [2^k,k,2^k-1] donc taille des messages = 2^k donc 16 ce qui correspond bien a notre matrice et aussi car on a dans Ghds une matrice identité de taille 4 et donc le rendement est de 0.5

2.

3.

4.1.

4.2. Sans implémenté la fonction et grâce aux recherches associées à la question 1 on sait que la distance de ce code de Hadamard est de 2^k-1 qui est ici égal à 8. On sait que d ≥ k +1 dans la détection d'erreur avec cette fois k = nombre d'erreurs donc il peut théoriquement détecter 7 erreurs. On sait aussi que d ≥ 2k +1 dans la correction d'erreur avec donc k = nombres d'erreurs donc le code de Hadamard peut corriger au maximum 3 erreurs.

4.3

5. Cette colonne renvoie tout le temps 0, mais le soucis c'est qu'elle ne produira pas le même encodage que celui de Hadamard si on l'enlève car cela causera un decalage et on sais pas où, du coup on pourrais ne pas décoder correctement le message.

6. On a toujours un k = 4. Et donc cela ne change pas la façon de faire et la distance sera toujours de 8 même si en théorie cela n'est peut-être pas vrai on pense que cela est toujours le cas.

8. En la calculant à la main on trouve cette matrice pour H de Gs :

0011|10000000000
0101|01000000000
0110|00100000000
0111|00010000000
1001|00001000000
1010|00000100000
1011|00000010000
1100|00000001000
1101|00000000100
1110|00000000010
1111|00000000001

Après quelques recherches on trouve que le code dual du simplexe de Hadamard est le code de Hamming

9. La vérification a été faite.

10.1. On peut d'abord créer toutes les erreurs pour savoir ce que donne le syndrome en cas d'erreur sur un bit précis.
Quand on multiplie H*E on se rend compte que cela renvoie le bit a l'opposé de l'erreur
Exemple pour l'erreur 0b0100000000000000 decode renvoie 00000000001 donc erreur sur le bit 1
idem pour 0b0010000000000000 decode renvoie 00000000010 et pour l'erreur 3 cela renvoie 000000000100
On en déduit donc que le reçoit la colonne correspondant a l'erreur mais a l'envers donc pour erreur 0 on reçoit la colonne 14 , et pour erreur 14 la colonne 0 de la matrice H.

10.2. Si on suit les éléments du cours on peut chercher d'abord pour une erreur sur un bit en fonction du message que l'on reçoit. On a donc besoin d'additionner l'erreur et le message et de voir quel syndrome cela renvoie. Exemple pour erreur = 0b0000000000000100 et message = 0b0100000000000000 le décodage donne : 0b01110001101.

On peut donc remarquer que cela correspond aux syndromes de l'erreur sur le bit 1 . On voit donc que notre message contient 2 erreurs car on a un syndrome qui est très reconnaissable mais qui est faux sur un bit et c'est donc là que l'on cherche le syndrome associé a une erreur au bit 9 de ce syndrome et on sait en regardant H qu'il s'agit du bit 13 qui est en erreur.

La meilleure solution serait pour nous de créer tout les codes possibles. C’est aàdire toutes les combinaisons d’erreurs possible pour obtenir un syndrome valide a chaque étape. Si on trouve un syndrome valide au premier décodage on a une erreur et on sais où la placé si on en a 3 alors il faut trouver le moyen de localiser correctement ces erreurs afin de les éliminer une par une.

10.3.

11. Dessin du registre a décalage :

P = 111101011001

```
*---
        |X0=1                 |X3    |X4           |X6             |X8    |X9     |X10  |X11
----->- + ->- + ->- + ->- + ->- + ->- + ->- + ->- + ->- + ->- + ->- + ->- + ->---->
dividende                                                                                   quotient
```

12.

13.

14.