

TP : MODELISATION DES CODES CORRECTEURS SOUS MATLAB SIMULINK

On souhaite modéliser le codage d'erreur (HAMMING(8,4)), à l'aide de Matlab/Simulink. Le message est constitué d'un seul caractère hexadécimal (4 bits).

Recopier le répertoire "modélisation codes correcteur" situé dans le cours moodle TELECOM NUMERIQUE section 2ENOC_TL_Telecom_num_2021 .

Partie I : Caractérisation code Hamming(8:4) .

I : Code hamming(7,4).

Message (4bits)	Code_ C(7;4) (4+3 bits redondance)
(m3,m2,m1,m0)	(c6,c5,c4,c3,c2,c1,c0)
	=(r2,r1,r0,m3,m2,m1,m0)
(msb first . . . lsb)	(msb first , , , , , , , , lsb)

I.1. Utiliser la commande matlab ; n=7;k=4 ;

Dans la fenêtre de commande matlab, saisir la ligne de commande : **[h,g,n,k]=hammgen(3)** .
Relever, sur votre compte rendu les matrices G et H qui permettent de coder les messages et de détecter les erreurs.

Note : Le code 'c' est obtenu en faisant $c = m \times G$.
La détection d'erreurs (syndrome 's') en faisant $s = c \times H$.

Ouvrir Matlab_R2010 dans vos applications locales

Dans Matlab....ouvrir le fichier «caractérisation_hamming.mdl ».

I.2. Etude de la partie « Caractérisation Hamming(7,4) ».

Vous travaillez dans la partie haute du schéma (« caractérisation Hamming(7.4)

Le message à émettre est un vecteur de 4 bits , fourni par un générateur aléatoire .
(«bernouilly binary generator » , bloc orange à gauche)

Travail à Réaliser :

- Configurer le bloc « canal de transmission »
(canal de transmission > error probability = 0.0). (bloc en orange dans le schéma).
Lancer la simulation (1 itération).**Vérifier la transmission sans erreur.**
- Modifier le taux d'erreur du canal de transmission pour faire apparaître des erreurs
(canal transmission > error probability 14%. soit un taux d'erreur=1/7)
- Lancer plusieurs fois la simulation (1000 itération chaque fois) et relever le nombre d'erreurs (bloc gris en fin de schema)
- A chaque itération, comparer le message émis avec celui reçu, et vérifier que le bloc « hamming(7,4) decoder » corrige bien une erreur mais pas au delà
Pour chaque itération, établir la correspondance entre le code émis (sans erreur), le syndrome et la position de l'erreur .

I.3. Etude de la génération du syndrome (bloc jaune).

Travail à Réaliser :

- Le syndrome est un vecteur de 3 bits $[s_1, s_2, s_3]$, image de la présence d'erreur et peut indiquer sa position dans le code. Il est le résultat du produit de c par la matrice H .
 - Décrire à l'aide d'équations logiques simples, la génération de ces 3 bits.

$$(s_1, s_2, s_3) = (c_6, c_5, c_4, c_3, c_2, c_1, c_0) \times H^T.$$
 - Compléter le bloc « calcul de syndrome » (en jaune).
- Lancer plusieurs fois la simulation (1 itération chaque fois) (canal transmission > error probability 12 %.)
- Pour chaque itération, comparer le message émis avec celui reçu, et vérifier que le bloc «calcul syndrome » indique bien la position de l'erreur dans H .
 - Dans ce cas, on limitera l'étude à une seule erreur de transmission.

II : Code Hamming(8,4), ajout d'un bit de parité

Etude de la partie « Caractérisation Hamming(7,4)+ bit parité=Hamming(8.4) ».

Message (4bits)

(m_3, m_2, m_1, m_0)

code_ C(8;4) (1+3 bits redondance+4)

$(c_7, c_6, c_5, c_4, c_3, c_2, c_1, c_0)$

$(p, r_2, r_1, r_0, m_3, m_2, m_1, m_0)$

Vous travaillez dans la partie basse du schéma (« caractérisation Hamming(7.4)+ bit parité

II.1. Configurer le bloc « canal de transmission 2» (canal de transmission 2> error probability = 0.0). (bloc en orange dans le schéma)

Travail à Réaliser :

- Lancer la simulation (1 itération). Vérifier la transmission sans erreur.
- Vérifier que les blocs « encodage(4 to 7) » et « décodage (7 to 4) » génèrent le même code que la fonction $[h, g, n, k] = \text{hammgen}(3)$, étudiée en partie I.1 .
On peut donc en conclure qu'on utilise les mêmes matrices G et H dans les deux parties du tp.

II.2. L'ajout du 8°bit (noté « p ») est fait dans le bloc « encodage parité» (en rouge).

Cela consiste à faire la somme des bits à « 1 » parmi les 7 bits du code.

Le bit « p » est codé de façon à toujours obtenir un nombre pair de bits à un dans le code 8 bits résultant (parité paire).

Travail à Réaliser :

- Lancer plusieurs fois la simulation (1 itération chaque fois).
 - Vérifier que les blocs « parité émise » et « parité reçue » génèrent le bon code de parité, (contrôler avec les bits affichés dans « Code_emis(7,4) »).
- La valeur du bit de parité émis et reçu doit être cohérente avec le nombre bits contenus dans le code .
- Vérifier la transmission sans erreur.**

II.3. Etude de la génération du syndrome

Etude du contrôle de parité (bocs jaunes). Le syndrome est le même vecteur de 3 bits $[s_1, s_2, s_3]$, issu des mêmes équations mises en place dans la partie I.3 de ce tp.

Méthode de détection des erreurs pour le codage de Hamming(8 :4)

- S'il n'y a aucune erreur, le syndrome est nul
- Si une erreur s'est produite sur les 7 premiers bits, le syndrome donne la position de l'erreur. L'existence d'un nombre impair d'erreur(s) est confirmée par le bit de parité (erreur de parité détectée).
- Si deux erreurs se sont produites :
Le syndrome n'est pas nul, et le bit de parité indique une parité exacte (nombre pair d'erreurs). Une retransmission est nécessaire.
- Si une erreur s'est produite sur le 8^e bit (bit de parité), l'absence d'erreur sur les 3 premiers éléments du syndrome (syndrome nul) permet de localiser l'erreur sur la parité, et le message est validé.

Travail à Réaliser :

- a. Compléter le bloc « calcul de syndrome 1 » (en jaune).
- b. Décrire à l'aide d'équations logiques simples, le décodage de la parité, qui consiste à faire la somme des bits à un dans le code 8 bits reçu.
- c. Compléter le bloc « détection parité » (en jaune).
- d. **Vérifier le bon fonctionnement.**

Partie II : Influence du codage de Hamming(8:4) sur le Bit Error Rate (A faire sous forme de compte rendu en marge du TL)

On souhaite caractériser l'influence du codage d'erreur (HAMMING(8,4)), sur les performances d'une liaison, à l'aide de Matlab/Simulink.

Le message est constitué d'un seul caractère hexadécimal (4 bits).

Lancer Matlab ... !

Message (4bits)

(m3,m2,m1,m0)

(msb first . . . lsb)

Code_ C(8;4) (1+3 bits redondance +4)

(c7,c6,c5,c4,c3,c2,c1,c0)

(p,r2,r1,r0,m3,m2,m1,m0)

(msb_first , , , , , , , , lsb)

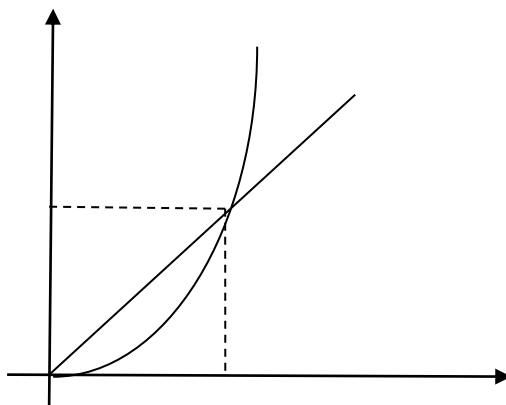
Dans Matlab....ouvrir le fichier «caractérisation_hamming.mdl ».

1-1 :Caractérisation Hamming(8:4):

Travail à Réaliser :

Ouvrir, le fichier «caractérisation_hamming.mdl» .

- Générateur aléatoire (Bernoulli binary) ,vérifiez qu'il est configuré comme suit;
Zero Probabilty = 0.5, Sample time =1 , sample per Frame = 4 (nbre de bit).
- Configurer le canal de transmission (Binary Symmetrical Chanel > error probability = 0.0). (bloc en orange dans les schémas)
- Lancer la simulation (1 itération)**Vérifier la transmission sans erreur.**
- Faites varier le taux d'erreur du canal de transmission (Binary Symmetrical Chanel > error probability entre 0% et 25% (au pas de 2%).
- Pour chaque valeur, lancer la simulation (au moins 1000 itérations) et relever;
«% d'erreur sur canal» «BER_Hamming» exprimé en %.
- Sous Open Office , ouvrir le fichier «resultat_tp_simulink.odc» reporter toutes vos mesures dans les colonnes prévues à cet effet.



1-2: Influence du code de Hamming(8,4) sur le BER du canal de transmission:

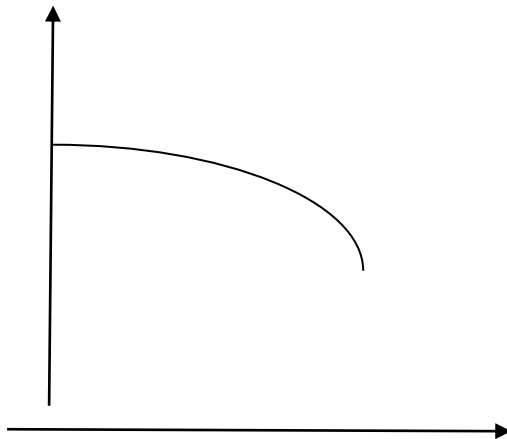
Ouvrir le fichier «canal_ber_hamming_bpsk.mdl»

- Le message à émettre est généré par un générateur aléatoire (Bernoulli binary)
Vérifier qu'il est configuré comme suit;
Zero Probability=0.5, Sample time=1, sample per Frame=4 (nbre de bit par message).
- Configurer les canaux de transmission AWGN > Signal to Noise ratio (SNR=+10dB)
- Lancer la simulation avec (10 000 itérations)
- Faites varier le taux d'erreur des 2 canaux de transmission (AWGN > Signal to Noise ratio (SNR)) +10dB et 0dB (au pas de 1dB).

Pour chaque valeur, lancer la simulation (10 000 itérations) et relev

«BER_canal_brut» «BER_Hamming»

- Sous Open Office, ouvrir le fichier «résultat_tp_simulink.odc» Reporter toutes vos mesures dans les colonnes prévues à cet effet



Exploitation des résultats PartieI:

- Analysez le $nbe_d'erreurs_détectées = f(nbr_d'erreur_sur_canal_brut)$,
en déduire la limite de bon fonctionnement du code Hamming(8:4).

Comment pourriez-vous expliquer ce résultat ?

Expliquer le rapport qui existe entre ce résultat et le fait de ne corriger qu'une erreur sur 7.

- Quelle est l'influence du codage de voie utilisé sur les performances du canal de transmission. Notion de gain de codage.

