



Simulation d'une Chaîne de Transmission au Format DVB-S

Professeur Encadrant : Benoit ESCRIG

Réalisé par :
GHALLABI Sidi Mohamed Raid
CHAN PENG Mael

Sciences du Numérique
Deuxième année - Parcours Réseaux
23 janvier 2025

Table des Matières

1	Introduction	2
2	Implantation du Modulateur/Démodulateur	3
2.1	Définition des Constantes et Paramètres	3
2.2	Donnez le schema de la chaine de transmission sur porteuse associee a une modulation de type QPSK, ainsi que celui de la chaine passe-bas equivalente.	4
2.2.1	Chaîne de transmission sur porteuse (QPSK)	4
2.2.2	Chaîne passe-bas equivalente	5
2.3	Suréchantillonnage	6
2.3.1	Condition de Shannon-Nyquist	6
2.3.2	Relation entre T_s , T_e et N_s	6
2.3.3	Valeur minimale de N_s	6
2.4	Détermination de l'efficacité spectrale	6
2.4.1	Calcul de l'efficacité spectrale	7
2.5	Calcul du Taux d'Erreur Binaire (TEB)	7
2.6	Taux d'Erreur Binaire Théorique (TEB)	7
2.7	Courbe	7
3	Ajout du Codage Canal	8
3.1	Introduction du Code Convolutif	8
3.1.1	Familiarisation avec le Codage Convolutif	8
3.1.2	Codage Convolutif (7, 1/2) pour le DVB-S	8
3.1.3	Effet du Poinçonnage sur le Codage Convolutif	9
3.2	Introduction du Code Bloc de Reed-Solomon	10
3.3	Introduction de l'Entrelaceur Convolutif	12
4	Conclusion	13
5	Bibliographie	14

1 Introduction

Le projet porte sur la simulation d'une chaîne de transmission conformément au standard DVB-S, en utilisant une modulation QPSK et un codage convolutif combiné au codage Reed-Solomon. Afin de renforcer la robustesse face aux erreurs en rafale, une technique d'entrelacement est intégrée entre les blocs de codage Reed-Solomon et convolutif. L'objectif est d'étudier les performances de cette chaîne sur un canal AWGN en termes de taux d'erreur binaire (TEB), avec et sans poinçonnage, et de comparer les performances obtenues avec des techniques de décodage hard et soft.

Objectifs

- Simuler une chaîne de transmission DVB-S conforme au standard, intégrant une modulation QPSK et évaluer ses performances en termes de TEB sur un canal AWGN.
- Intégrer le codage convolutif et analyser son impact sur les performances avec un décodage SOFT/HARD, avec/sans poinçonnage.
- Intégrer le codage Reed-Solomon pour corriger les erreurs et analyser son impact sur les performances.
- Insérer l'entrelacement pour améliorer les performances du codage.

2 Implantation du Modulateur/Démodulateur

Dans cette partie, nous implémentons un modulateur et démodulateur QPSK avec une mise en forme par un filtre racine de cosinus surélevé (SRRC). Le signal suréchantillonné est simulé sur un canal AWGN (Additive White Gaussian Noise). Voici une explication détaillée des étapes du code :

2.1 Définition des Constantes et Paramètres

Les paramètres suivants sont définis pour la simulation :

- **M** : L'ordre de la modulation QPSK, donc $M = 4$.
- **nbits** : Le nombre total de bits à transmettre, fixé à $188 \times 8 = 1504$ bits. Cela correspond à la taille totale d'un paquet de transport MPEG-2, comme le montre la [figure 1](#) prise de DVB-S, incluant 1 octet de synchronisation et 187 octets de données utiles, chaque octet contenant 8 bits.

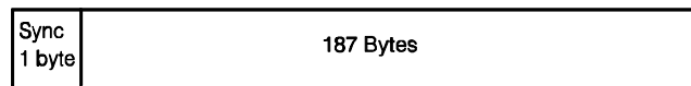


Figure 3a) MPEG-2 transport MUX packet

FIGURE 1 – Structure d'un paquet de transport MPEG-2
PAGE :11 de Norme DVB-S

- **alpha** : Le roll-off du filtre SRRC, fixé à 0.35 (conforme à la norme DVB-S).
- **Ns** : Le facteur de suréchantillonnage, fixé à 5.
- **span** : La durée du filtre SRRC, fixée à 10.
La valeur de 'span' doit être comprise entre 6 et 10, comme expliqué aux séances de TPs.

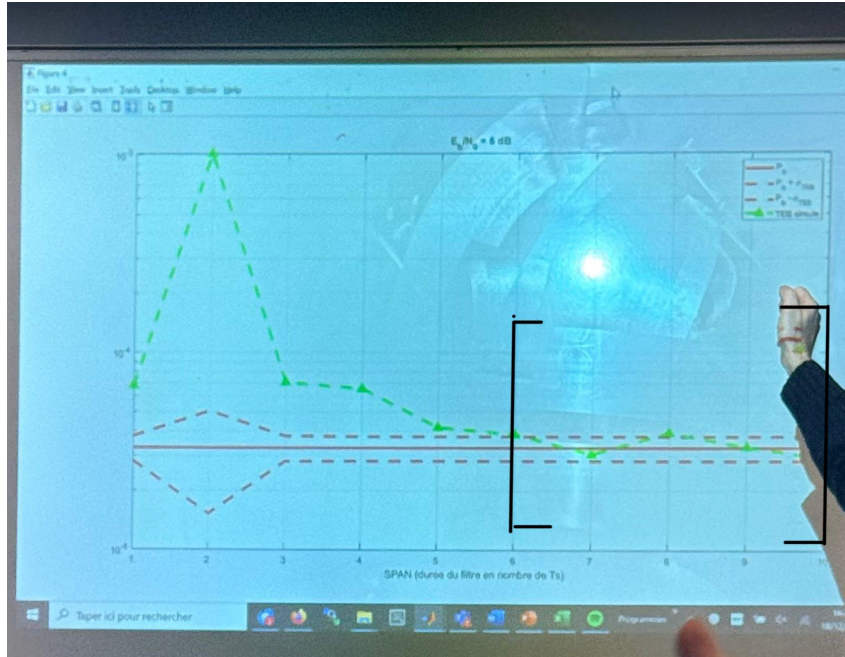


FIGURE 2 – Choix de span

$$\begin{aligned} span &\in [6, 10] \\ span \times Ns + 1 &\text{ périodes d'échantillonnage.} \end{aligned}$$

2.2 Donnez le schema de la chaine de transmission sur porteuse associee a une modulation de type QPSK, ainsi que celui de la chaine passe-bas equivalente.

2.2.1 Chaîne de transmission sur porteuse (QPSK)

La chaîne de transmission sur porteuse pour une modulation QPSK comprend les étapes suivantes :



FIGURE 3 – Schéma de la chaîne de transmission sur porteuse QPSK

1. **Génération des bits** : Les bits d'information sont générés aléatoirement grâce à **randi.m**. Ces bits représentent les données à transmettre.
2. **Mapping QPSK** : Les bits sont regroupés par deux (puisque QPSK transmet 2 bits par symbole) et mappés sur des symboles complexes selon la constellation QPSK. Par exemple :
 - Les bits 00 sont mappés sur le symbole complexe $1 + j$.
 - Les bits 01 sont mappés sur le symbole complexe $-1 + j$.
 - Les bits 10 sont mappés sur le symbole complexe $1 - j$.

— Les bits 11 sont mappés sur le symbole complexe $-1 - j$.

$$x_e(t) = I(t) + jQ(t)$$

3. **Filtrage de mise en forme** : Les symboles QPSK sont filtrés par un filtre en racine de cosinus surélevé (SRRCF) pour limiter la bande passante du signal et éviter l'interférence entre symboles (ISI). Le filtre SRRCF est conçu avec un roll-off de 0.35, conformément au standard DVB-S.

`h = rcosdesign(alpha, span, sps, 'sqrt');`

4. **Modulation sur porteuse** : Les symboles filtrés sont modulés sur une porteuse haute fréquence (RF) en utilisant une modulation en quadrature (I et Q). Cette étape consiste à multiplier les symboles complexes par une porteuse sinusoïdale pour les transposer dans la bande de fréquence désirée.
5. **Transmission sur le canal** : Le signal modulé est transmis sur un canal de communication, souvent modélisé comme un canal à bruit additif blanc gaussien (AWGN). Ce canal ajoute du bruit au signal, ce qui simule les perturbations rencontrées dans un environnement de transmission réel.

2.2.2 Chaîne passe-bas équivalente

La chaîne passe-bas équivalente est une représentation simplifiée de la chaîne de transmission. Les étapes principales sont :

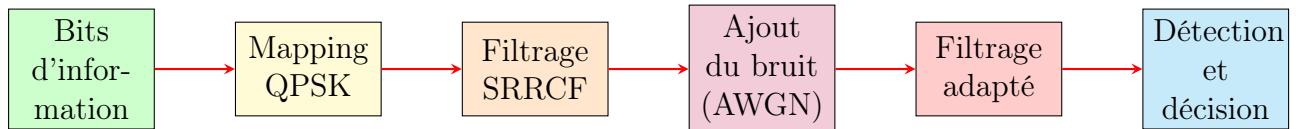


FIGURE 4 – Schéma de la chaîne de transmission QPSK

1. **Génération des bits** : Les bits d'information sont générés aléatoirement grâce à `randi.m`.
2. **Mapping QPSK** : Les bits sont mappés sur des symboles complexes.

$$x_e(t) = I(t) + jQ(t)$$

3. **Filtrage de mise en forme** : Les symboles sont filtrés par un filtre en racine de cosinus surélevé (SRRCF) pour limiter la bande passante. (`filter.m`)

`h = rcosdesign(alpha, span, sps, 'sqrt');`

4. **Ajout du bruit** : Le bruit complexe (AWGN) est ajouté au signal filtré pour simuler les effets du canal de communication. La puissance du bruit est calculée en fonction du rapport signal à bruit par bit E_b/N_0 , qui varie de -4 dB à 4 dB.

$$n_e(t) = n_I(t) + jn_Q(t) \quad \text{avec} \quad S_{nI}(f) = S_{nQ}(f) = N_0$$

5. **Filtrage adapté** : Le signal reçu est filtré par un filtre adapté (également un SRRCF). (`filter.m`)
6. **Détection et décision** : Les symboles reçus sont détectés et convertis en bits.

2.3 Suréchantillonnage

Les symboles QPSK sont suréchantillonnés en insérant $N_s - 1$ zéros entre chaque symbole par la fonction `kron.m`. Cette opération permet d'augmenter la résolution temporelle du signal pour le filtrage.

2.3.1 Condition de Shannon-Nyquist

$$F_e \geq 2 \cdot F_s$$

2.3.2 Relation entre T_s , T_e et N_s

$$T_s = N_s \cdot T_e$$

$$F_e = N_s \cdot F_s$$

2.3.3 Valeur minimale de N_s

$$N_s \cdot F_s \geq 2 \cdot F_s$$

$$\boxed{N_s \geq 2}$$

2.4 Détermination de l'efficacité spectrale

L'efficacité spectrale d'une chaîne de transmission est définie par :

$$\eta = \frac{R_b}{B}$$

où :

- η : l'efficacité spectrale en bits/s/Hz,
- R_b : le débit binaire total (en bits/s),
- B : la largeur de bande nécessaire pour transmettre le signal (en Hz).
- **Largeur de bande nécessaire** : Pour une modulation QPSK, la largeur de bande B est donnée par :
$$B = (1 + \alpha) \cdot R_s$$
- **Le débit symbole R_s** : est donné par :

$$R_s = \frac{R_b}{\log_2(M)}$$

2.4.1 Calcul de l'efficacité spectrale

L'efficacité spectrale η est donnée par :

$$\eta = \frac{\log_2(M)}{1 + \alpha}$$

L'efficacité spectrale théorique de la chaîne de transmission sans codage canal est :

$$\eta = \frac{2}{1,35} \approx 1,48 \text{ bits/s/Hz}$$

$$\boxed{\eta \approx 1.48 \text{ bits/s/Hz}}$$

2.5 Calcul du Taux d'Erreur Binaire (TEB)

Le TEB est calculé en comparant les bits transmis et les bits reçus. Le nombre de bits erronés est divisé par le nombre total de bits transmis pour obtenir le TEB simulé.

2.6 Taux d'Erreur Binaire Théorique (TEB)

Le TEB théorique pour une modulation QPSK est calculé en utilisant la fonction Q (fonction d'erreur complémentaire) `qfunc.m`. Les résultats simulés sont comparés aux valeurs théoriques pour valider le bon fonctionnement de la chaîne de transmission.

$$\text{TEB} = Q\left(\sqrt{\frac{2E_b}{N_0}}\right)$$

2.7 Courbe

La courbe de TEB est tracée en échelle logarithmique.

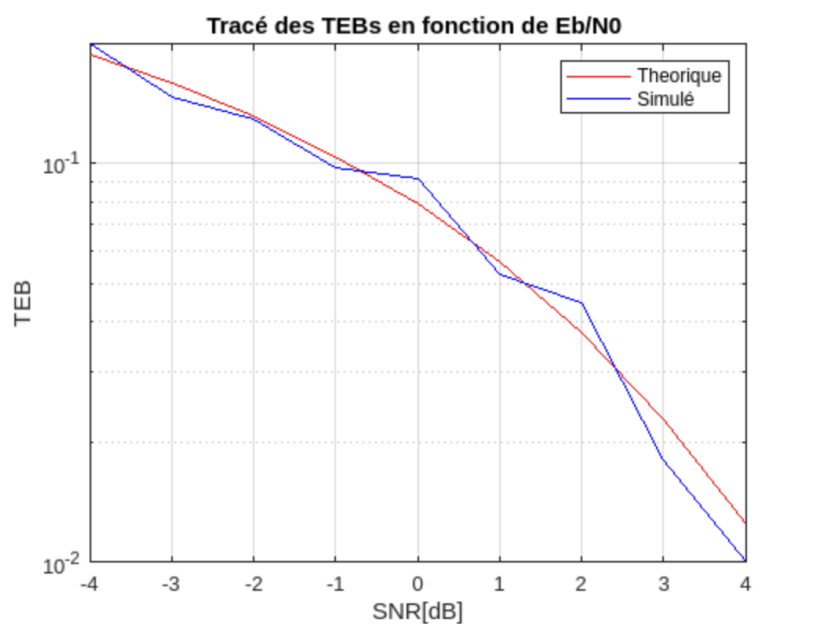


FIGURE 5 – Comparaison entre le TEB théorique et simulé

Cette figure (figure 5) montre le bon fonctionnement de notre chaîne, car les deux courbes sont proches.

3 Ajout du Codage Canal

3.1 Introduction du Code Convolutif

3.1.1 Familiarisation avec le Codage Convolutif

J'ai implémenté un code convolutif $(3, 1/2)$ avec les polynômes générateurs $g_1 = 5_{\text{oct}}$ et $g_2 = 7_{\text{oct}}$. J'ai utilisé la fonction `poly2trellis.m` de MATLAB pour générer le treillis associé à ces polynômes. Ensuite, j'ai encodé une séquence binaire d'entrée avec `convenc.m` et analysé les bits codés en sortie. Pour décoder, j'ai appliqué l'algorithme de Viterbi via `vitdec.m`. Comme vous pouvez voir sur la figure 6, les bits décodés correspondaient bien à la séquence d'origine, validant ainsi la mise en œuvre du code convolutif et du processus de décodage.

```
>> codeconvolutif
Message binaire d'entrée :
    0    1    0    1    1    0    1

Message encodé (code convolutif) :
    0    0    1    1    0    1    0    0    1    0    1    0    0    0

Message décodé (après le décodage avec Viterbi) :
    0    1    0    1    1    0    1

Nombre d'erreurs entre le message original et le message décodé :
    0
```

FIGURE 6 – code convolutif $(3, 1/2)$

3.1.2 Codage Convolutif $(7, 1/2)$ pour le DVB-S

J'ai implémenté un code convolutif $(7, 1/2)$ conforme au standard DVB-S, avec les polynômes générateurs $g_1 = 171_{\text{oct}}$ et $g_2 = 133_{\text{oct}}$. Ce code a été intégré à un bloc modulateur/démodulateur pour simuler une transmission sur un canal bruité. J'ai testé deux modes de décodage : **hard** (décision dure) et **soft** (décision souple), en utilisant l'algorithme de Viterbi via la fonction `vitdec.m` de MATLAB.

Pour comparer les performances, j'ai mesuré les Taux d'Erreurs Binaires (TEB) pour différents rapports E_b/N_0 .

Les résultats illustrés dans la figure 7 montrent que le décodage **soft** offre de meilleures performances que le décodage **hard**, avec un TEB plus faible pour un même E_b/N_0 . Cela s'explique par le fait que le décodage soft se base sur des **probabilités** ou des métriques de fiabilité des bits reçus, plutôt que sur des décisions binaires comme dans le cas du décodage hard. En utilisant ces informations supplémentaires, le décodeur soft peut mieux distinguer les erreurs et prendre des décisions plus précises.

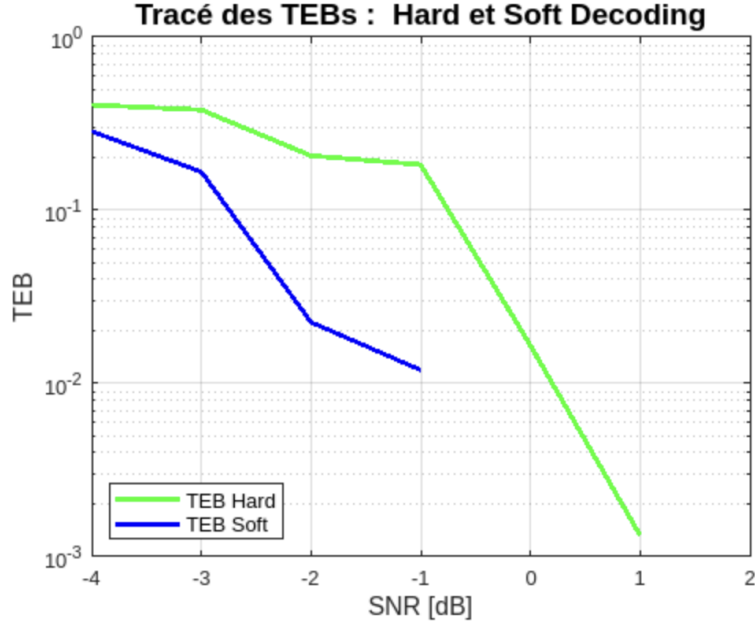


FIGURE 7 – Comparaison entre le TEB obtenu avec un decodage Hard et Soft

3.1.3 Effet du Poinçonnage sur le Codage Convolutif

J'ai utilisé une matrice de poinçonnage $P = [1\ 1\ 0\ 1]$ pour transformer un code convolutif de taux $R = 1/2$ en un code de taux $R = 2/3$. Le poinçonnage permet de supprimer certains bits de la séquence codée selon un motif prédéfini, augmentant ainsi le débit utile. J'ai comparé les Taux d'Erreurs Binaires (TEB) obtenus avec et sans poinçonnage pour un décodage **soft**. Les résultats montrent que le poinçonnage entraîne une légère augmentation du TEB, car il réduit la redondance du code, mais il permet d'augmenter le débit de transmission.

Table 2: Punctured code definition

Original code			Code rates									
K	G1 (X)	G2 (Y)	1/2		2/3		3/4		5/6		7/8	
			P	dfree	P	dfree	P	dfree	P	dfree	P	dfree
7	171 _{oct}	133 _{oct}	X: 1 Y: 1 I=X ₁ Q=Y ₁	10	X: 1 0 Y: 1 1 I=X ₁ Y ₂ Y ₃ Q=Y ₁ X ₃ Y ₄	6	X: 1 0 1 Y: 1 1 0 I=X ₁ Y ₂ Q=Y ₁ X ₃	5	X: 1 0 1 0 1 Y: 1 1 0 1 0 I=X ₁ Y ₂ Y ₄ Q=Y ₁ X ₃ X ₅	4	X: 1 0 0 0 1 0 1 Y: 1 1 1 1 0 1 0 I=X ₁ Y ₂ Y ₄ Y ₆ Q=Y ₁ Y ₃ X ₅ X ₇	3

NOTE: 1 = transmitted bit
0 = non transmitted bit

Avantages du poinçonnage :

- Augmentation du débit utile sans modifier la structure du code convolutif.
- Flexibilité pour adapter le taux de codage aux besoins du système.

Inconvénients du poinçonnage :

- Réduction de la redondance, ce qui diminue la capacité de correction d'erreurs.
- Légère dégradation des performances en termes de TEB, surtout à faible E_b/N_0 .

En ce qui concerne le décodage **hard**, il s'avère inefficace en cas de poinçonnage. En effet, le décodage hard ne tient pas compte de la fiabilité des bits reçus, ce qui le rend

incapable de compenser la perte d'information due au poinçonnage. Le décodage **soft**, en revanche, exploite des informations probabilistes pour reconstruire les bits manquants, ce qui le rend plus robuste dans ce contexte. Cela explique pourquoi le décodage soft est préférable pour les systèmes utilisant le poinçonnage.

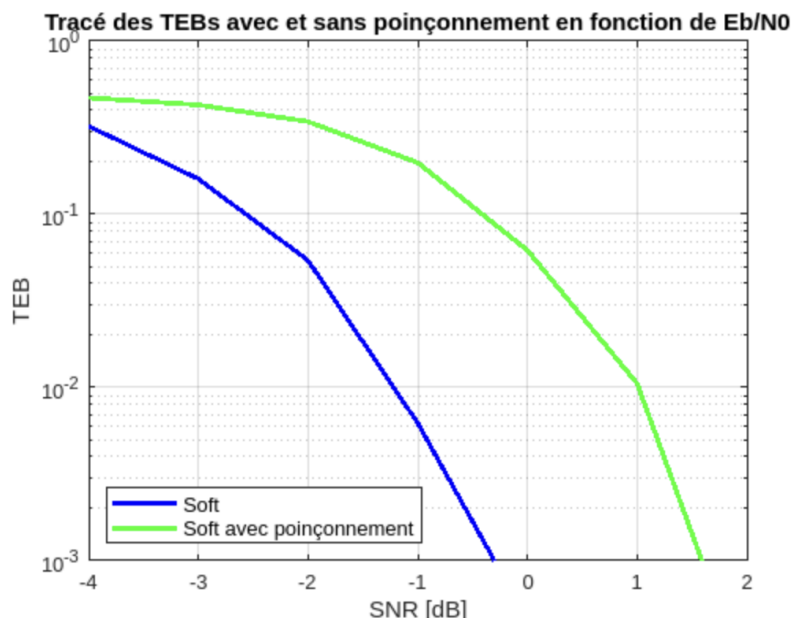


FIGURE 8 – Comparaison entre le TEB obtenu avec un décodage Soft avec et sans poinçonnage

Les courbes de TEB en fonction de E_b/N_0 de la [figure 8](#) montrent que le poinçonnage dégrade légèrement les performances par rapport au code non poinçonné, mais cette dégradation est acceptable compte tenu de l'augmentation du débit. Le décodage soft permet de limiter cette dégradation grâce à son utilisation efficace des informations probabilistes.

3.2 Introduction du Code Bloc de Reed-Solomon

Pour satisfaire au standard **DVB-S**, j'ai ajouté un code de **Reed-Solomon (RS)** à la chaîne de transmission existante (bloc modulateur/démodulateur + codage convolutif sur canal AWGN). Le code utilisé est **RS(204, 188)**, une version raccourcie du code **RS(255, 239)**. Ce code est capable de corriger jusqu'à 8 erreurs par bloc de 188 octets, ce qui améliore significativement la robustesse de la transmission.

Mise en œuvre avec MATLAB

Pour générer les matrices de codage et de décodage, j'ai utilisé les fonctions suivantes de MATLAB :

- `comm.RSEncoder.m` : Pour encoder les données avec le code RS.
- `comm.RSDecoder.m` : Pour décoder les données reçues.

Le codage et le décodage ont été réalisés en utilisant la fonction `step` de MATLAB. Voici une partie du code :

```

% Paramètres du code RS
n = 204; % Longueur du bloc codé
k = 188; % Longueur du bloc d'information

% Création de l'encodeur et du décodeur RS
encoder_rs = comm.RSEncoder(N, K, BitInput=true);
decoder_rs = comm.RSDecoder(N, K, BitInput=true);

% codage et décodage

bits = randi([0, 1], 1, nbits); % Bits aléatoires
bits_rs = step(encoder_rs, bits. '); % Codage RS
decodedsoft_rs_final = step(decoder_rs, decodedsoft_rs. '); % Décodage RS

```

Comparaison des Taux d'Erreurs Binaires (TEB)

J'ai comparé les TEB obtenus **sans** et **avec** le code de Reed-Solomon. Les résultats illustrés dans la [figure 9](#) montrent que l'ajout du code RS réduit significativement le TEB, en particulier dans des conditions de faible rapport signal à bruit (E_b/N_0). Cela s'explique par la capacité du code RS à corriger les erreurs résiduelles après le décodage convolutif.

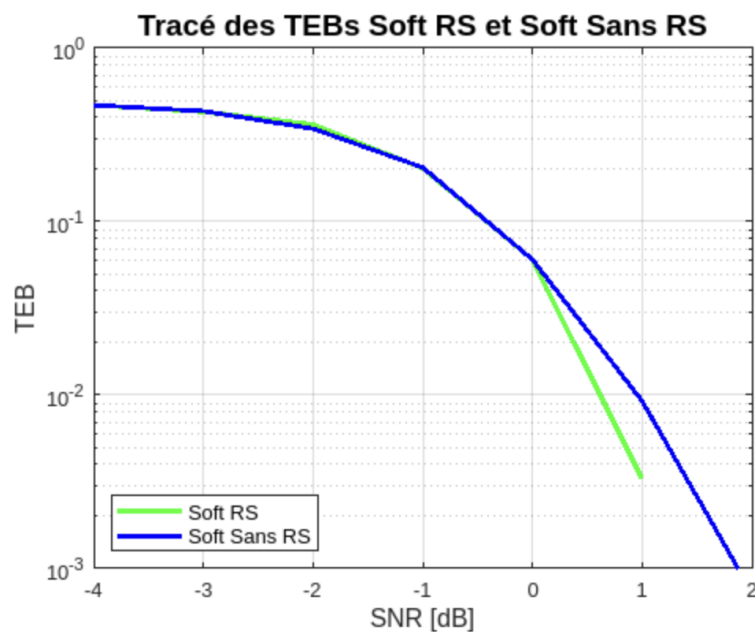


FIGURE 9 – Comparaison entre le TEB obtenu avec et sans codage Reed Solomon

Intérêts de la concaténation de codes de types différents

La concaténation d'un code convolutif et d'un code de Reed-Solomon présente plusieurs avantages :

- **Correction d'erreurs améliorée** : Le code convolutif est efficace pour corriger les erreurs aléatoires, tandis que le code RS est particulièrement adapté pour corriger les erreurs en rafales. Ensemble, ils offrent une protection robuste contre différents types d'erreurs.

- **Flexibilité** : Le code RS peut être ajusté pour corriger un nombre spécifique d'erreurs, ce qui permet d'adapter la robustesse de la transmission aux conditions du canal.
- **Performance accrue** : La combinaison des deux codes permet d'atteindre des TEB très faibles, ce qui est essentiel pour des applications critiques comme la diffusion satellite (DVB-S).

Pourquoi le décodage hard est inefficace avec le code RS

Le décodage **hard** est inefficace avec le code RS car il ne tient pas compte de la fiabilité des symboles reçus. Le code RS fonctionne mieux avec un décodage **soft**, qui utilise des informations probabilistes pour identifier et corriger les erreurs. Cela permet de maximiser la capacité de correction du code RS, en particulier dans des environnements bruités.

3.3 Introduction de l'Entrelaceur Convolutif

Pour améliorer les performances du codage, le standard **DVB-S** propose d'insérer un **entrelaceur convolutif** entre le code de Reed-Solomon et le code convolutif. L'entrelaceur a pour objectif de disperser les erreurs en rafales, qui peuvent survenir sur le canal de transmission, afin de les rendre plus faciles à corriger par les codes correcteurs d'erreurs.

Implémentation avec MATLAB

Pour implanter l'entrelaceur convolutif, j'ai utilisé les fonctions suivantes de MATLAB :

- `convintrlv.m` : Pour l'entrelacement des symboles.
- `convdeintrlv.m` : Pour le désentrelacement des symboles.

On utilise les fonctions `bitstooctets` et `octetstobits` pour que l'entrelacement s'applique sur les **symboles du code Reed-Solomon (octets)**, et non sur les bits individuels.

Comparaison des Taux d'Erreurs Binaires (TEB)

J'ai ajouté l'entrelacement à la chaîne de transmission existante (canal AWGN avec codage de Reed-Solomon et codage convolutif) et comparé les TEB obtenus **sans** et **avec** entrelacement.

Nous n'avons pas réussi la partie de l'entrelacement : la **figure 10** présente une contradiction. Le résultat devait montrer que l'entrelacement réduit le TEB. Car l'entrelaceur a la capacité de disperser ces erreurs (erreurs en rafale), permettant aux codes correcteurs de les corriger plus efficacement.

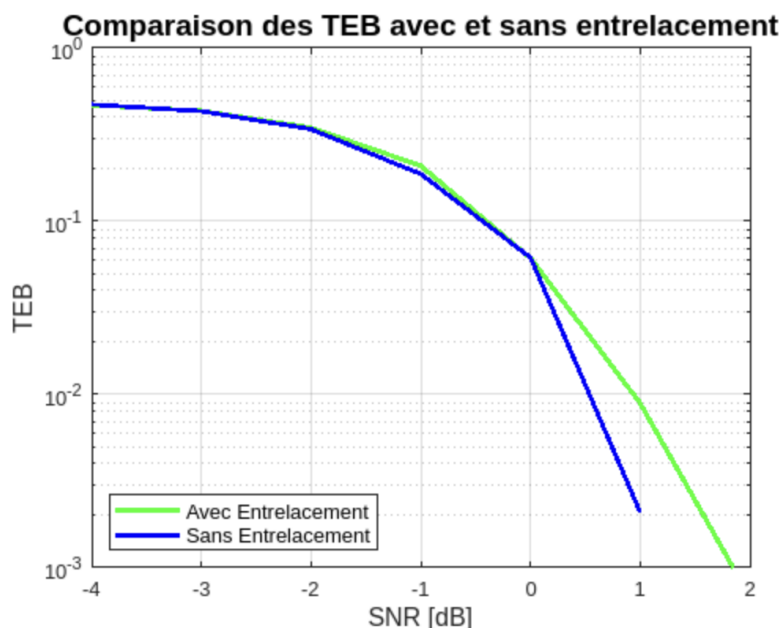


FIGURE 10 – Comparaison entre le TEB obtenu avec et sans entrelacement (malheureusement fausse)

Avantages de l'entrelacement

L'entrelacement permet d'améliorer les performances du codage canal de plusieurs manières :

- **Dispersion des erreurs en rafales** : Les erreurs en rafales, qui peuvent survenir sur un canal bruité, sont dispersées dans le temps. Cela les rend plus faciles à corriger par les codes correcteurs d'erreurs.
- **Robustesse** : En combinant l'entrelacement avec des codes comme Reed-Solomon et le code convolutif, la transmission devient plus robuste aux perturbations du canal.
- **Amélioration du TEB** : L'entrelacement réduit le TEB en permettant une correction plus efficace des erreurs, en particulier dans des environnements à fort bruit.

L'entrelacement est efficace car il transforme les erreurs en rafales (qui sont corrélées et difficiles à corriger) en erreurs isolées (qui sont plus faciles à corriger). Cela est particulièrement utile dans les systèmes de communication où les erreurs en rafales sont fréquentes, comme dans les transmissions par satellite (DVB-S).

4 Conclusion

Ce projet a permis de simuler une chaîne de transmission conforme au standard **DVB-S**, en intégrant une modulation QPSK, un codage convolutif, un codage de Reed-Solomon et un entrelacement convolutif. L'objectif principal était d'étudier les performances de cette chaîne sur un canal AWGN en termes de Taux d'Erreurs Binaires (TEB), avec et sans poinçonnage, et de comparer les techniques de décodage **hard** et **soft**.

Les résultats obtenus montrent que :

- **Le codage convolutif** améliore significativement les performances de la transmission, en particulier avec un décodage **soft** qui exploite des informations probabilistes pour corriger les erreurs. Le poinçonnage, bien qu'il augmente légèrement le TEB, permet d'augmenter le débit utile tout en maintenant une bonne robustesse.
- **Le code de Reed-Solomon** ajoute une couche supplémentaire de protection contre les erreurs, en particulier les erreurs en rafales. Sa combinaison avec le codage convolutif offre une correction d'erreurs robuste.
- **L'entrelacement convolutif** disperse les erreurs en rafales, les rendant plus faciles à corriger par les codes correcteurs. Cela améliore significativement les performances de la transmission, en particulier dans des environnements bruités.

En conclusion, la chaîne de transmission simulée répond aux exigences du standard **DVB-S**, offrant une transmission robuste et efficace grâce à la combinaison de techniques de codage et d'entrelacement. Les résultats démontrent l'importance de l'utilisation conjointe de codes de types différents (convolutif et bloc) et de techniques de décodage adaptées (soft) pour maximiser les performances dans des environnements de transmission réels.

5 Bibliographie

- **EN 300 421 V1.1.2 (1997-08)** : European Standard (Telecommunications series). *Digital Video Broadcasting (DVB) ; Framing structure, channel coding and modulation for 11/12 GHz satellite services*. European Telecommunications Standards Institute (ETSI), 1997. Support mis à notre disposition sur Moodle
- **MATLAB Documentation** : *Communications Toolbox*. MathWorks, 2023. Disponible sur : <https://www.mathworks.com/help/comm/>.
- **Boucheret, M.-L.** : *Support de cours sur le codage canal*. Slides des cours dispensées par Madame Marie-Laure Boucheret.