



Modélisation du comportement d'un modem sur un signal numérique.

Hammi Kamal et Basile Gros

Département Sciences du Numérique - Première année
2020-2021

Contents

1	Introduction	3
2	Modem de fréquence - Démodulation par filtrage	3
2.1	Construction du signal modulé en fréquence	3
2.1.1	Génération du signal NRZ	3
2.1.2	Génération du signal modulé en fréquence	4
2.2	Canal de transmission à bruit additif, blanc et Gaussien	5
2.3	Démodulation par filtrage	5
2.3.1	Synthèse du filtre passe-bas	6
2.3.2	Synthèse du filtre passe-haut	6
2.3.3	Filtrage, tracés, commentaires sur les résultats obtenus	7
2.3.4	Détection d'énergie	8
2.3.5	Erreur dans la démodulation	8
2.4	Application de la recommandation V21	9
3	Modem de fréquence V21 - Démodulateur FSK	9
3.1	Démodulateur FSK - Contexte de synchronisation idéale	9
3.2	Démodulateur FSK avec gestion d'une erreur de synchronisation de phase porteuse	10
4	Conclusion	11
5	Références	11

List of Figures

1	Signal modulé en fréquence	3
2	Comparaison entre la densité spectrale de puissance du signal NRZ et sa valeur théorique.	4
3	Comparaison entre la densité spectrale de puissance du signal modulé en fréquence et sa valeur théorique.	5
4	Démodulation par filtrage.	6
5	Réponse impulsionnelle du filtre passe bas.	6
6	Réponse impulsionnelle du filtre passe bas.	7
7	Réponse impulsionnelle du filtre passe bas.	7
8	Réponse impulsionnelle du filtre passe bas.	8
9	Démodulation FSK. Synchronisation supposée idéale.	9
10	Démodulation FSK. Synchronisation supposée idéale.	10
11	Démodulation FSK - Gestion d'une erreur de phase porteuse.	11

1 Introduction

L'objectif de ce projet est de simuler un modem suivant la norme V21. Pour cela, nous avons créé un modulateur FSK (Frequency Shift Key) qui module les 0 et les 1 avec deux fréquences F_0 et F_1 dans un signal sinusoïdal. Le signal est ensuite perturbé par un bruit blanc gaussien avant d'être démodulé.

Nous avons construit trois démodulations différentes que nous avons comparé dans le cadre des recommandations de modulation de la norme V21.

- La première filtre le signal pour extraire une des fréquences.
- La deuxième méthode suppose que le signal est synchrone, c'est à dire qu'on en connaît la phase à tout instant, et en fait le produit scalaire avec les cosinus ayant servi à le créer.
- La troisième méthode de démodulation abandonne cette hypothèse et projette le signal sur les espaces des cosinus de fréquence F_0 et F_1 et de phase quelconque.

2 Modem de fréquence - Démodulation par filtrage

2.1 Construction du signal modulé en fréquence

La première étape du projet consiste à réaliser la modulation de fréquence, i.e. transformer l'information binaire à transmettre en un signal modulé en fréquence (exemple sur la figure 1).

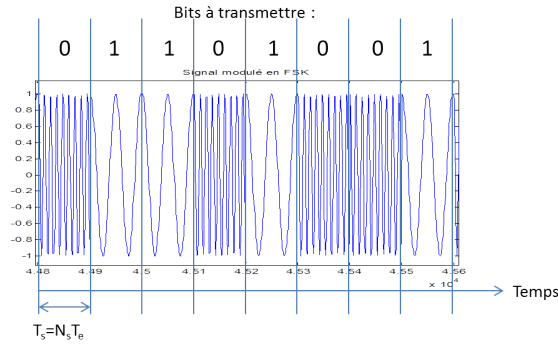


Figure 1: Signal modulé en fréquence

Le signal discret modulé en fréquence $x(t)$ est généré de la manière suivante :

$$x(t_i) = (1 - NRZ(t_i)) \times \cos(2\pi F_0 t + \phi_0) + NRZ(t_i) \times \cos(2\pi F_1 t + \phi_1) \quad (1)$$

où $NRZ(t)$ est un signal discret de type NRZ polaire formé à partir de la suite de bits à transmettre en codant les 0 et les 1 par des niveaux 0 et 1 de durée T_s secondes. Les phases ϕ_0 et ϕ_1 sont des variables aléatoires indépendantes uniformément réparties sur $[0, 2\pi]$. Les fréquences que nous utiliserons dans un premier temps seront $F_0 = 6000\text{Hz}$ et $F_1 = 2000\text{Hz}$ pour nous permettre une plus grande lisibilité et une facilité à distinguer les fréquences pour corriger nos erreurs.

2.1.1 Génération du signal NRZ

Le nombre N_s d'échantillons du signal modulé nécessaire pour coder un bit est :

$$N_s = \frac{T_s}{T_e} = \frac{F_e}{F_s}$$

avec F_s le débit du signal en bits par secondes, F_e la fréquence d'échantillonnage voulue pour le signal modulé en fréquence et T_e la période correspondante.

La génération du signal NRZ consiste à faire un produit tensoriel à une dimension (produit de Kronecker) entre le vecteur des bits d'entrée et un vecteur de taille N_s rempli de 1.

La densité de puissance du signal NRZ est :

$$S_{NRZ}(f) = \frac{1}{4}T_s \text{sinc}^2(\pi f T_s) + \frac{1}{4}\delta(f) \quad (2)$$

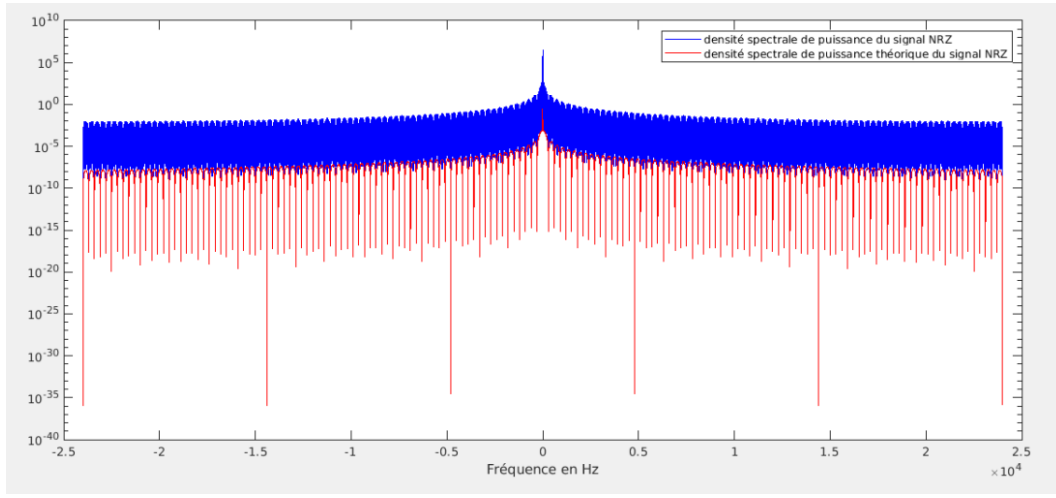


Figure 2: Comparaison entre la densité spectrale de puissance du signal NRZ et sa valeur théorique.

Dans la figure 2, la valeur théorique et la valeur calculée de la densité spectrale de puissance sont très similaires. La différence d'échelle est due aux coefficients présents lors du calcul du périodogramme. La formule que nous avons trouvée est donc la bonne.

2.1.2 Génération du signal modulé en fréquence

Pour générer le signal modulé en fréquence, nous avons simplement implémenté l'équation 1 à partir du signal NRZ précédent. La densité spectrale de puissance théorique du signal modulé en fréquence est :

$$S_{NRZ}(f) \times \frac{\delta(f - f_1) + \delta(f - f_0) + \delta(f + f_1) + \delta(f + f_0)}{4} \quad (3)$$

Dans la figure 3, la valeur théorique et la valeur calculée de la densité spectrale de puissance sont très similaires. C'est une différence de puissance similaire à celle de la figure 2.

Ce calcul théorique nous a posé des problèmes à cause d'une erreur sur l'autocorrélation mais après avoir relu en détail et avec l'aide de notre professeur référent, nous avons compris que notre erreur était dans le calcul du produit par le conjugué.

De plus, nous avons eu du mal à placer nos diracs aux bonnes fréquences sur matlab. Cela était dû au zero-padding employé dans le périodogramme qui nous empêchait de pointer aux bonnes fréquences, qui n'étaient plus des multiples du pas entre deux points.

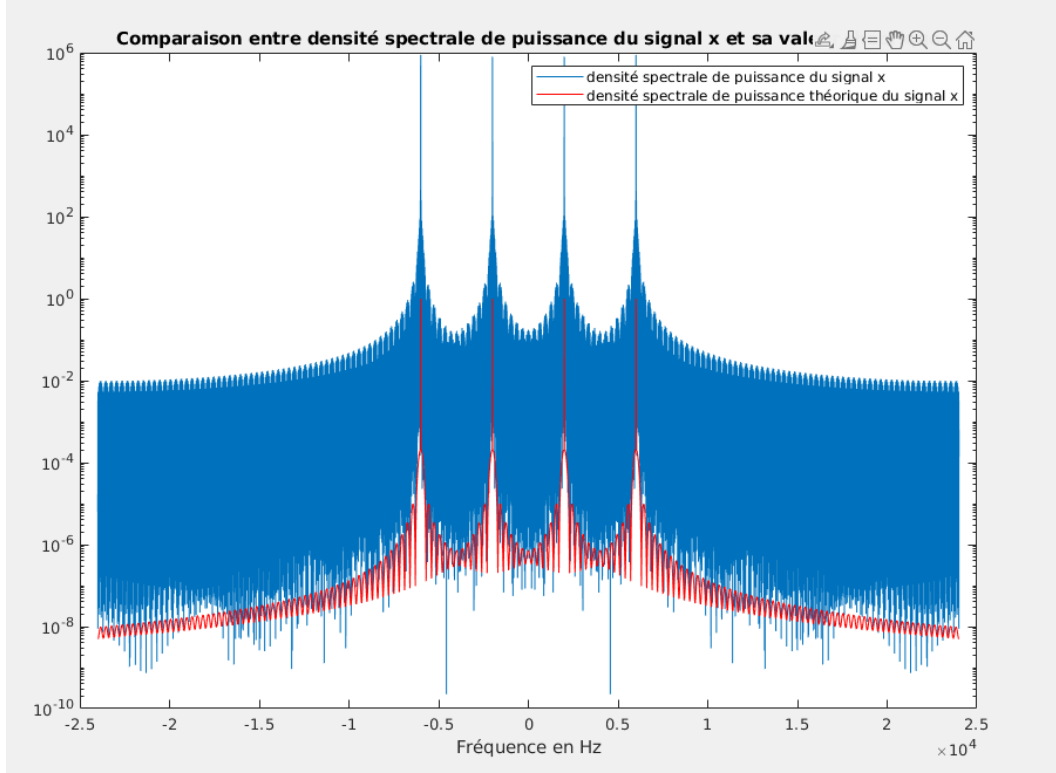


Figure 3: Comparaison entre la densité spectrale de puissance du signal modulé en fréquence et sa valeur théorique.

2.2 Canal de transmission à bruit additif, blanc et Gaussien

Nous allons considérer que le canal de propagation ajoute au signal émis un bruit que l'on suppose blanc et Gaussien et qui modélise les perturbations introduites.

La puissance du bruit Gaussien à ajouter devra être déduite du rapport signal sur bruit (SNR : Signal to Noise Ratio) souhaité pour la transmission donné en dB :

$$SNR_{dB} = 10 \log_{10} \frac{P_x}{P_b}$$

où P_x représente la puissance du signal modulé en fréquence et P_b la puissance du bruit ajouté. Sauf exception, nous prendrons une valeur de $SNR_{dB} = 50$ pour tracer nos figures ce qui est d'après le sujet une bonne valeur pour tester un modem.

2.3 Démodulation par filtrage

La figure 4 présente le récepteur implémenté pour retrouver, à partir du signal modulé en fréquence bruité, le message binaire envoyé.

Un filtre passe-bas permet d'éliminer les morceaux de cosinus à la fréquence $F_0 = 6000\text{Hz}$, tandis qu'un filtre passe-haut permet de filtrer les morceaux de cosinus à la fréquence $F_1 = 2000\text{Hz}$. Une détection d'énergie réalisée tous les T_s secondes permet de récupérer, à partir des signaux filtrés, les bits 0 et 1 transmis.

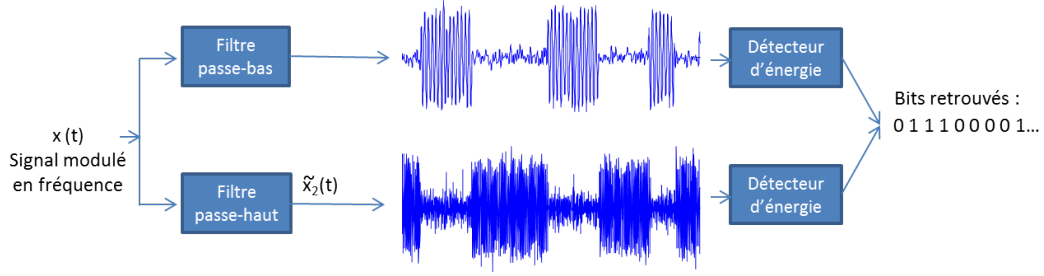


Figure 4: Démodulation par filtrage.

2.3.1 Synthèse du filtre passe-bas

Nous avons choisi un ordre de filtre égal à 255 car nous travaillons sur des images à 256 niveaux de gris et nous avons besoin d'un ordre impair. Le filtre passe bas a une fréquence de coupure $F_c = \frac{F_0 + F_1}{2F_e}$ qui est la fréquence moyenne des deux fréquences utilisées qui a été normalisée. La réponse fréquentielle du filtre est une fonction porte de support $2F_C$. L'équation de sa réponse impulsionnelle est donc :

$$2F_C \text{sinc}(2\pi t F_C)$$

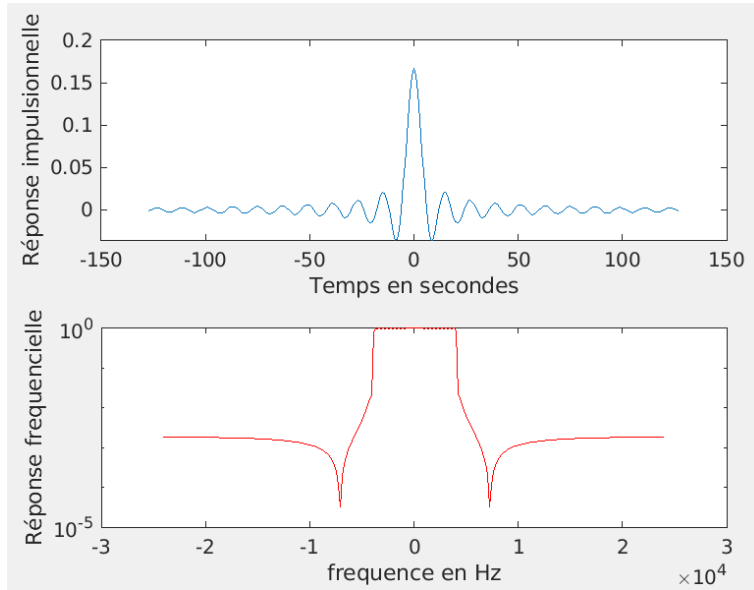


Figure 5: Réponse impulsionnelle du filtre passe bas.

2.3.2 Synthèse du filtre passe-haut

Le filtrage passe haut a été fait de manière similaire au filtre passe bas. Sa réponse fréquentielle étant $H_{IPH}(\tilde{f}) = 1 - H_{IPB}(\tilde{f})$, on obtient par transformée de Fourier inverse l'équation de sa réponse impulsionnelle :

$$\delta(0) - 2F_C \text{sinc}(2\pi t F_C)$$

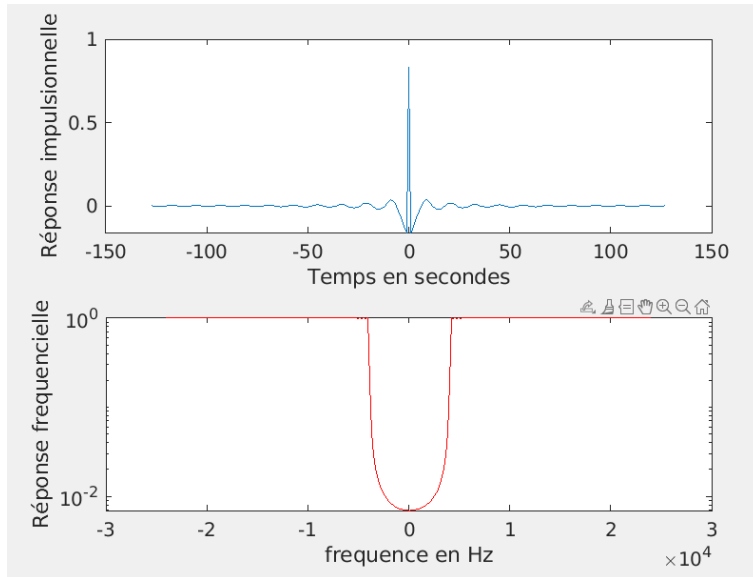


Figure 6: Réponse impulsionnelle du filtre passe bas.

2.3.3 Filtrage, tracés, commentaires sur les résultats obtenus

La première étape consiste à regarder si les filtres vont nous donner le bon résultat. Pour cela, il suffit de superposer leur réponse fréquentielle avec la densité spectrale de puissance du signal à filtrer, et de regarder si seule la fréquence voulue est gardée.

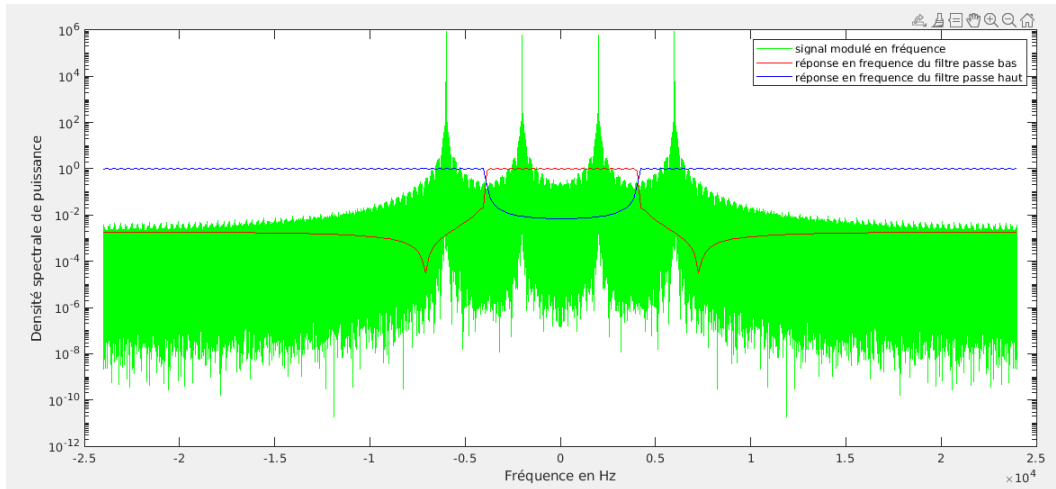


Figure 7: Réponse impulsionnelle du filtre passe bas.

La figure 7 nous confirme que le filtre passe haut laisse seulement passer les pics à 6000Hz et -6000Hz des bits 0 tandis que le filtre passe bas filtre ces pics mais laisse passer les pics à 2000Hz et -2000Hz des bits 1.

Avant de filtrer, il faut bien prendre en compte le décalage induit par l'ordre du filtre. Pour cela, il faut que ajouter à la fin du signal à filtrer autant de zéros que l'ordre du filtre, pour que toutes les valeurs soient traitées. De même, après le filtrage, il faut décaler le signal filtré de l'ordre du filtre pour retrouver le signal initial, car l'ordre du filtre induit aussi un décalage.

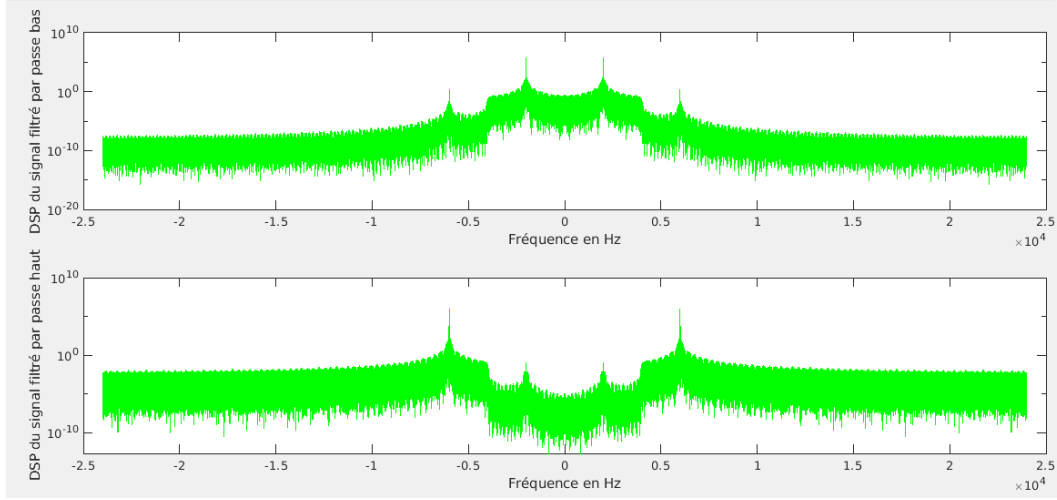


Figure 8: Réponse impulsionnelle du filtre passe bas.

Après avoir filtré le signal, la figure 8 nous montre que les pics de puissance filtrés ont bien une puissance diminuée, multipliée par la valeur du filtre en fréquence. Le filtrage a donc bien fonctionné. Pour récupérer le signal final, on choisit un des deux signaux filtrés, en sachant que l'autre permettra d'obtenir des résultats similaires.

2.3.4 Détection d'énergie

Pour décoder le signal, il faut calculer l'énergie de chaque bit. Pour cela, le signal est séparé en intervalles de N_s points dont on calcule le carré de la norme. Après avoir calculé toutes ces carrés, on les sépare par un seuil égal à la moyenne entre les valeurs minimales et maximales. Toutes les énergies supérieures à ce seuil seront décodées en 1 et les énergies inférieures à ce seuil en 0.

$$Si ||bit||^2 = \sum_{n=1}^{N_s} x_n^2 > K \text{ alors bit 1 sinon bit 0}$$

$$avec K = \frac{Energie\ maximale + Energie\ minimale}{2}$$

La raison pour laquelle nous calculons des carrés de normes et pas des normes est que cela évite un calcul de racine carrée, très coûteux s'il est fait des très nombreuses fois sur de grandes images.

2.3.5 Erreur dans la démodulation

Puisque nous possédons le signal d'entrée du modem, nous pouvons le comparer avec le signal obtenu après démodulation. Nous calculons un taux d'erreur binaire, variant entre 0 et 1, la distance de Hamming entre le signal original et le signal reconstruit, normalisée par la longueur de ces signaux.

Tous nos tests sont d'abord réalisés sur des séquences de 1500 bits générés selon une loi uniforme chaque fois que nous lançons la simulation. Chaque test est répété au moins une dizaine de fois. Puis, nous utilisons des images fournies avec le sujet pour tester des vecteurs de bits plus importants.

Pour un SNR_{dB} de 50dB, le taux d'erreur obtenu ici est de 0, mais si on descend ce rapport à 1dB, le taux d'erreur passe à 0.029 en moyenne. La méthode des filtrages est donc très efficace dans ce cas ci.

2.4 Application de la recommandation V21

La norme V21 préconise des fréquences $F_0 = 1180\text{Hz}$ et $F_1 = 980\text{Hz}$. Comme la différence entre ces fréquences sont plus proches d'un facteur 10 par rapport à la différence entre les fréquences que nous avons utilisées jusque là. Cela entraîne des problèmes dans le filtrage des fréquences. Même en l'absence de bruit, ou avec un SNR_{dB} très élevé (nous sommes monté jusqu'à $1000dB$) le taux d'erreur binaire n'est pas nul, oscillant entre 0.005 et 0.012.

Après avoir examiné les différents graphes, nous avons remarqué que la fréquence de coupure des filtres est légèrement décalée à cause de la discrétisation, comme on le voit sur la figure 9. Nous en avons conclu que cette erreur venait de la fréquence d'échantillonnage F_e trop faible pour ces fréquences, la réponse fréquentielle du filtre a une résolution trop faible, comme montré dans la figure 9.

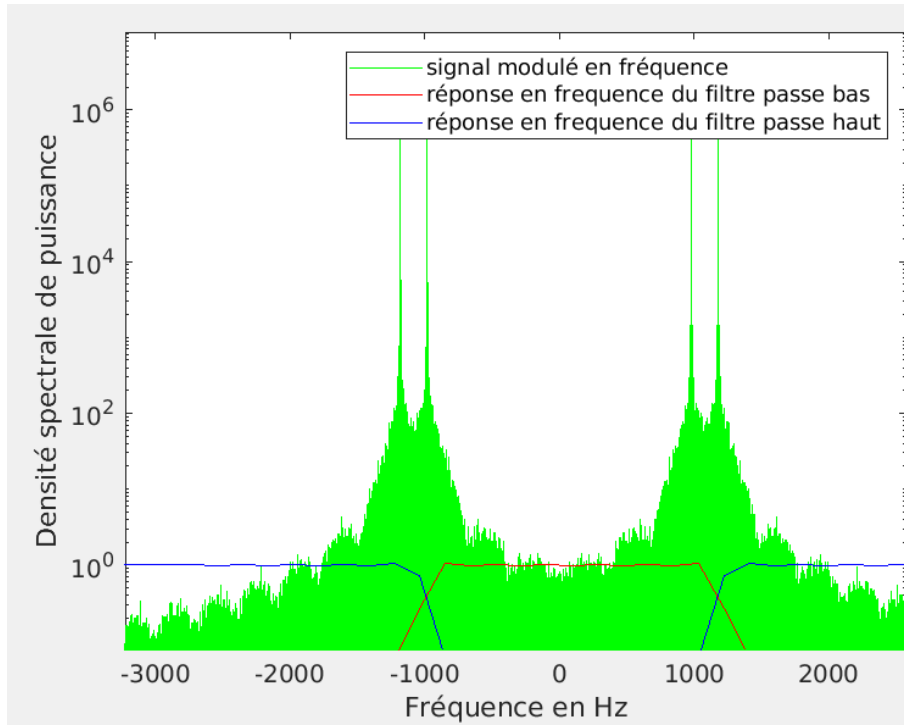


Figure 9: Démodulation FSK. Synchronisation supposée idéale.

Changer les paramètres du filtre comme la fréquence de coupure, l'ordre du filtre ou la fenêtre de troncature n'y a rien changé. Néanmoins, changer la fréquence F_e (en la doublant par exemple) permet de retrouver une erreur nulle.

3 Modem de fréquence V21 - Démodulateur FSK

3.1 Démodulateur FSK - Contexte de synchronisation idéale

La figure 10 présente le récepteur modifié implémenté afin de retrouver, à partir du signal modulé en fréquence suivant la recommandation V21, le message binaire envoyé.

Dans cette démodulation, on suppose que l'on connaît les phases des cosinus utilisés pour créer le signal NRZ. On va alors utiliser un produit scalaire pour projeter nos bits sur les droites des cosinus utilisés. En utilisant le produit scalaire :

$$\langle f, g \rangle = \int_0^{T_s} f g$$

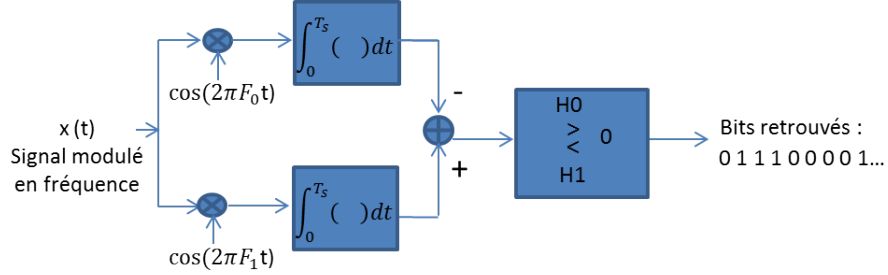


Figure 10: Démodulation FSK. Synchronisation supposée idéale.

on a les résultats :

$$| \langle \cos(2\pi F_0 t + \Phi_0), \cos(2\pi F_0 t + \Phi_0) \rangle | = \left| \frac{\sin(4\pi F_0 T_s) + 4\pi F_0 T_s - \sin(2\Phi)}{8\pi F_0} \right| \approx \frac{T_s}{2} = 0.00165$$

qui est du même ordre de grandeur que

$$| \langle \cos(2\pi F_1 t + \Phi_1), \cos(2\pi F_1 t + \Phi_1) \rangle |$$

et

$$\begin{aligned} & | \langle \cos(2\pi F_0 t + \Phi_0), \cos(2\pi F_1 t + \Phi_1) \rangle | = \\ & \left| \frac{(f_1 - f_0) \sin(\Phi_1 + \Phi_0 + (2\pi f_1 + 2\pi f_0) T_s) + (f_1 + f_0) \sin(\Phi_1 - \Phi_0 + (2\pi f_1 - 2\pi f_0) T_s)}{4\pi f_1^2 - 4\pi f_0^2} - \right. \\ & \left. \frac{(f_1 - f_0) \sin(\Phi_1 + \Phi_0) + (f_1 + f_0) \sin(\Phi_1 - \Phi_0)}{4\pi f_1^2 - 4\pi f_0^2} \right| \leq 3 \times 10^{-5} \end{aligned}$$

Ainsi, les différences entre les normes et les produits scalaires croisés sont suffisamment importantes pour qu'on puisse utiliser la même détection d'énergie que pour le filtrage, cette fois en obtenant un taux d'erreur binaire nul pour $SNR_{dB} = 50dB$. La comparaison à faire est la même que pour un filtre passe haut. Sinon le taux d'erreur est de 1, ce qui revient à inverser tous les bits et montre une démodulation sans erreurs aussi.

Le seul problème que nous avons rencontré a été de bien comprendre comment gérer la phase entre les différentes fréquences à la modulation.

3.2 Démodulateur FSK avec gestion d'une erreur de synchronisation de phase porteuse

Dans cette partie, on ne suppose plus qu'on connaît les phases des cosinus. Nous modélisons cela en introduisant une différence de phase dans le signal d'entrée qui n'est pas répercutée sur le signal de sortie. Alors, nous obtenons un taux d'erreur binaire proche de 0.5 signifiant que toute l'information est perdue. Puisque les phases ne sont plus synchronisées, le produit scalaire ne fonctionne plus, l'entrée et la sortie ne sont pas sur la même demi-droite.

Pour palier ce problème, il suffit de projeter les bits sur les espaces à deux dimensions des cosinus de fréquences F_0 et F_1 de phase quelconque, dont une base orthogonale est le cosinus et le sinus de cette fréquence. Ainsi, la désynchronisation de la phase ne fait plus sortir de l'espace de projection, et le problème est résolu. C'est cette opération qui est représentée à la figure 11. On retrouve donc bien un taux d'erreur nul pour un $SNR_{dB} = 50dB$.

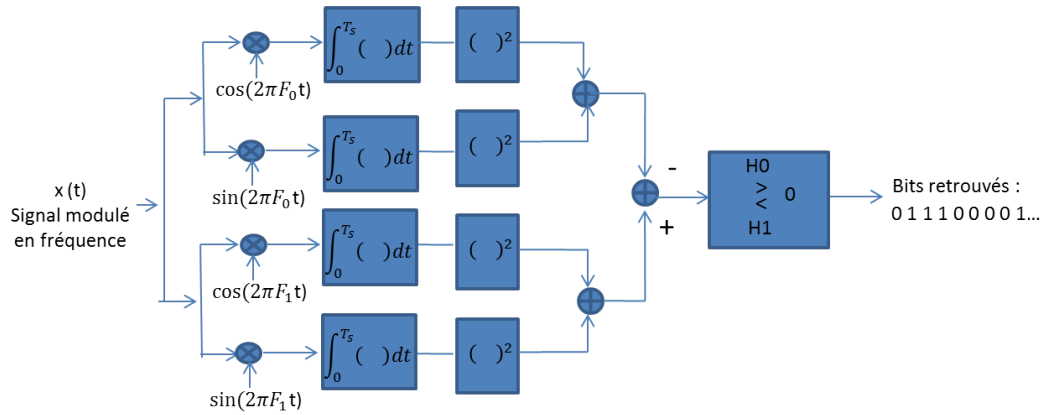


Figure 11: Démodulation FSK - Gestion d'une erreur de phase porteuse.

4 Conclusion

Lorsque l'on transmet un signal entre un émetteur et un récepteur, ce signal est bruité. Le bruit rencontré le plus fréquemment est un bruit blanc Gaussien.

Dans le cas d'une modulation de fréquence, la méthode la plus évidente pour démoduler ce signal bruité est de le filtrer pour séparer les deux fréquences de modulation. Ce procédé fonctionne bien quand le bruit est faible par rapport au signal, ou quand les fréquences sont éloignées, mais il nécessite une fréquence d'échantillonnage très élevée pour fonctionner avec des fréquences rapprochées comme celles de la norme V21. Cette accroissement de la fréquence d'échantillonnage est coûteuse en puissance de traitement et en espace de stockage du signal.

Il est alors possible d'utiliser la méthode de démodulation FSK (Frequency Shift Key) qui consiste à projeter le signal sur les espaces des fonctions cosinus ayant servi lors de la modulation. Si la phase de ses cosinus est connue, seules deux intégrales, une pour chaque fréquence, sont nécessaires pour retrouver le signal de départ. Mais ce n'est pas le cas de la majorité des transmissions, où la phase des signaux de modulation est inconnue.

Il faut alors utiliser deux intégrales par fréquence, augmentant le temps de calcul nécessaire pour traiter le signal, mais rendant possible sa lecture en asynchrone. lorsqu'il est ainsi désynchronisé. C'est pour cela que cette dernière méthode a été retenue dans le cas de norme V21.

5 Références

- Le site Mathworks pour comprendre à utiliser des fonctions plus avancées de Matlab : <https://fr.mathworks.com>
- Le site de calcul formel en ligne Wolframalpha pour vérifier nos calculs : <https://www.wolframalpha.com>